Quality-oriented control of video delivery over wireless channels

Nesrine Changuel

Université Paris-Sud 11
Faculté des Sciences dOrsay

THÈSE DE DOCTORAT

Spécialité : **Physique**

Ecole Doctorale: **Sciences et Technologies de l'Information des Télécommunications et des Systèmes**

présentée par

Nesrine Changuel

Sujet:

Régulation de la qualité lors de la transmission de contenus vidéo sur des canaux sans fils

le **14 Décembre 2011**

devant le jury composé de:

M. Frederic Dufaux, Télécom ParisTech, CNRS, LTCI	Présidente du jury
M. Moncef Gabbouj, Université de Technologie de Tampere	Rapporteur
M. Markus Rupp, Université de Technologie de Vienne	Rapporteur
M. Hugues Mounier, LSS, CNRS, Supélec	Examinateur
M. Michel Kieffer, LSS, CNRS, Supélec	Directeur de thèse
M. Bessem Sayadi, Alcatel Lucent Bell-Labs France	co-directeur de thèse

A mes très chers parents, à mes brillantes soeurs,
A mon cher et tendre époux.

Remerciements

Ce rapport conclue ma thèse au sein d'Alcatel-Lucent Bell Labs et le laboratoire des signaux et systèmes. C'est donc avec beaucoup d'émotion que je m'adresse à tous ceux qui m'ont accompagné aux cours de ces trois années pour leur présenter mes très sincères remerciements.

Je tiens tout d'abord à exprimer ma profonde gratitude à mon directeur de thèse Michel Kieffer et le remercier pour son aide, sa disponibilité, et son soutien tout au long de cette thèse. Grâce à ses conseils, sa rigueur et ses compétences techniques, il a su orienter la thèse vers des approches innovantes combinant l'aspect pratique avec des concepts théoriques. Travailler sous sa direction était une expérience humaine et professionnelle très enrichissante.

Mes vifs remerciements vont également à mon co-directeur de thèse Bessem Sayadi, qui m'a d'abord fait confiance et ensuite guidée et conseillée tout au long de ces trois années. J'ai pu bénéficier de ses compétences techniques et théoriques, ainsi que de son aptitude à déceler les orientations susceptibles de mener à des résultats scientifiques intéressants.

I would like to express my gratitude to Mihaela van der Schaar who welcomed me to her research group as a trainee and gave me the opportunity to learn and acquire new scientific skills. I have learned a lot from her and this has added much value to this thesis.

Je souhaite remercier également toutes les personnes que j'ai pu côtoyer et avec lesquelles j'ai pu collaborer durant mes trois années de thèse chez Alcatel-Lucent Bell-Labs, et plus spécifiquement, toutes les équipes rattachées à Laurent Thomas. Mes plus vifs remerciements s'adressent surtout à Sylvaine, Frédéric, Marie-Line, et Vinod, pour l'aide précieuse qu'ils m'ont apporté durant cette aventure.

J'adresse mes sincères remerciements aux membres du jury pour le temps qu'ils ont consacré pour étudier mon travail ainsi que pour leurs commentaires très positifs.

La thèse a parfois été un moment difficile pour mes proches. Elle est très preneuse de temps. Pour cela, un grand merci à mon cher Nizar, qui partage ma vie et avec qui la vie prend tout son sens et l'avenir tout son intérêt. Qu'il soit assuré de ma gratitude et de l'importance du rôle qu'il a joué (et qu'il continue à jouer) pour moi.

Merci aussi à mes soeurs qui chacune de sa façon a contribué au succès de cette thèse.

Enfin, je remercie du fond du coeur et avec un grand amour ma très chère maman et mon très cher papa qui n'ont jamais cessé de croire en moi pendant toutes mes années d'études et qui m'ont toujours encouragé à aller de l'avant. Le plus beau soutien est la fierté que je lis dans leurs yeux, et qui est aussi ma plus belle récompense. Je leur dédie cette thèse ainsi que tout ce que j'ai pu réussir dans ma vie.

Résumé

Le développement simultané de terminaux mobiles multimédia (smartphones, tablettes) et de réseaux daccès offrant des débits élevés conduit à une explosion du trafic lié aux contenus multimédia. Cette croissance nécessite un partage efficace des ressources radio entre fournisseurs de contenus (dans le cas de la diffusion) ou entre récepteurs (dans le cas de services de vidéo à la demande). Cette thèse propose des outils de partage équitable des ressources en termes de qualité des contenus multimédia reçus et de délai de transmission dans les deux contextes précédents. La variété des compromis débit-distorsion des contenus multimédia est exploitée à cet effet.

Dans un premier temps, une solution centralisée de contrôle conjoint du débit de codage et de transmission de plusieurs programmes transmis sur un même canal est considérée. Lobjectif est de fournir des flux de qualités similaires avec des variations limitées, tout en assurant des délais de transmission comparables. Ce problème est résolu dans un premier temps en synthétisant une commande prédictive à laide doutils doptimisation sous contrainte.

Dans un second temps, seule lallocation de bande est centralisée, le contrôle des caractéristiques de compression de chaque flux est réalisé de manière distribuée. Le contrôleur centralisé ne renvoie que le niveau de remplissage des tampons associés à chaque flux aux fournisseurs de contenus distants. Une stratégie de régulation des débits de codage est alors mise en place par ces fournisseurs, de manière à réguler le niveau en bits ou en images des tampons. La stabilité de ce système de régulation couplé est étudiée en détail.

Enfin, loptimisation inter-couches dune chaîne de transmission de contenus multimédia scalable est considérée. Ce problème est formulé dans le contexte de la programmation dynamique. Lorsque des modèles de complexité raisonnable sont considérés et avec des caractéristiques du système bien connues, des solutions optimales peuvent être obtenues. Des techniques dapprentissage sont mises en oeuvre lorsque le système nest que partiellement connu, par exemple, lorsque létat du canal de transmission parvient avec du retard à lorgane de commande.

Mots clés:

codage, commande prédictive, contrôle distribué, optimisation inter-couches, optimisation sous contraintes, programmation dynamique, qualité vidéo, régulation.

Abstract

Due to the emergence of new generation mobile terminals and media streaming services, data traffic on mobile networks is continuously exploding.

Despite the emergence of standards such as LTE, resources still remain scarce and limited. Thus, efficiently sharing resources among broadcasters or between unicast receivers connected to the same base station is necessary. An efficient resources allocation, where a fair received video quality between users and an equal transmission delay are achieved, is targeted. To that end, the wide variety of rate-distortion trade-offs of multimedia contents is exploited.

First, a centralised joint encoding and transmission rate control of multiple programs sharing the same channel is considered. A satisfactory and a comparable video quality among the transmitted programs, with limited variations, as well as a comparable transmission delay are targeted. The problem is solved using constrained optimization tools.

Second, only the bandwidth allocation control is centralised, the control of the encoding rate of each stream is carried in a distributed manner. By modelling the problem as a feedback control system, the centralised bandwidth allocation is required to feed back only the buffer level to its associated remote content provider. The equilibrium and stability issues are addressed for both bit and image level buffer control.

In the case of simple unicast connection, a cross-layer optimization of scalable video delivery over a wireless channel is performed. The optimization problem is cast in the context of dynamic programming. When models with small size of the state space are considered and when the system characteristics are known, optimal solutions can be obtained. When the system is partially known, for example, when the state of the channel reaches the controller with delay, learning techniques are implemented.

Keywords:

distributed control, dynamic programming, constrained optimization problem, cross-layer optimization, foresighted control, regulation, video coding, video quality

Publications and patents

Journal papers

1. Changuel, N. and Sayadi, B. and Kieffer, M., **Predictive encoder and buffer control for statistical multiplexing of multimedia content**, submitted to IEEE Transaction on Broadcasting, August 2011.

2. Changuel, N. and Sayadi, B. and Kieffer, M., **Statistical multiplexing of video programs**, IEEE Vehicular Technology Magazine, pp. 62 - 68, no. 3, September, 2009.

Conference papers

1. Changuel, N. and Sayadi, B. and Kieffer, M., **Statistical multiplexing with distributed video coders control**, Submitted to IEEE ICC 2012.

2. Changuel, N. and Sayadi, B. and Kieffer, M., **Statistical multiplexing of distributed video streams**, Gretsi, pp. 1 - 4, September 2011, Bordeaux.

3. Changuel, N. and Sayadi, B. and Kieffer, M., **Delay-Sensitive Statistical Multiplexing of Multimedia Contents With Time-Varying Channel Conditions**, Proc. World Wireless Research Forum, April 2011, Qatar.

4. Changuel, N. and N. Mastronarde, N. and van der Schaar, M. and Sayadi, B. and Kieffer, M., **Adaptive scalable layer filtering process for video scheduling over wireless network based on MAC buffer management**, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), pp. 2352 - 2355, May, 2011, Prague.

5. Changuel, N. and N. Mastronarde, N. and van der Schaar, M. and Sayadi, B. and Kieffer, M., **End-to-end Stochastic Scheduling of Scalable Video Over Time Varying Channels**, ACM Multimedia, pp. 731 - 734, October, 2010, Florence.

6. Changuel, N. and Sayadi, B. and Kieffer, M., **Dependent encoder and buffer control for statistical multiplexing of multimedia contents**, IEEE Globecom , pp. 1 - 6, December, 2010, Miami.

7. Changuel, N. and Sayadi, B. and Kieffer, M., **H.264/AVC Inter-Frame Rate-Distortion dependency analysis based on independent regime**

switching AR models, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), pp. 914 - 917, March, 2010, Dallas.

8. Changuel, N. and Sayadi, B. and Kieffer, M., **Rate and Distortion Model for Efficient Statistical Multiplexing of Digital Video Programs**, Proc. World Wireless Research Forum, Paris, May 2009, paper selected to be published in IEEE Vehicular Technology Magazine 2009.

9. Changuel, N. and Sayadi, B. and Kieffer, M., **Predictive control for efficient statistical multiplexing of digital video programs**, Proc. IEEE Packet Video, pp. 1 - 9, May, 2009, Seattle.

White paper

1. Changuel, N. and Leprovost,Y. and Sayadi,B., **Scalable Video Coding: principles and benefits**, Vx/NetN/T/08/0093, Alcatel Lcent Bell Labs, 2008.

Patents

1. Changuel, N. and Sayadi, B. and Kieffer, M., **Statistical multiplexing of distributed video streams**, 2011, European Patent, 809506, filed 16/05/2011, Alcatel-Lucent.

2. Changuel, N. and Sayadi, B. and Kieffer, M., **Video packet filtering algorithm based on only MAC buffer management**, 2010, European Patent, 808627, filed 25/11/2010 Alcatel-Lucent.

3. Changuel, N. and Sayadi, B. and Kieffer, M., **Scheduling Method For Data Streaming in a Telecommunication Network Infrastructure**, 2010, European Patent, 808083, filed 28/09/2010, Alcatel-Lucent.

4. Changuel, N. and Sayadi, B. and Kieffer, M., **Method for allocating bit rates to video programs transmitted within a fixed rate channel**, 2009, European Patent, 9305287.6, filed 06/04/2009, Alcatel-Lucent.

5. Changuel, N. and Sayadi, B. and Kieffer, M., **Joint Encoder and Buffer Regulation for Statistical Multiplexing of Multimedia Contents**, 2009, European Patent, 9290584, filed 24/07/2009, Alcatel-Lucent.

List of acronyms

AVC	Advanced Video Coding
BL	Base layer
CBR	Constant Bit Rate
DP	Dynamic Programming
DVB-SH	Digital Video Broadcasting-Satellite Handheld
EL	Enhancement Layer
ET	Encoding Trials
GoP	Group of Pictures
IRS-AR	Independent Regime-Switching AutoRegressive
MAC	Medium Access Control
MB	MacroBlock
MBMS	Multimedia Broadcast Multicast Services
MDP	Markov Decision Process
MGS	Medium Grain Scalability
MIMO	Multiple Input Multiple Output
MOS	Mean Opinion Score
MPEG	Moving Picture Experts Group
N-SMC	N-State Markov Chain
PDU	Packet Data Units
PSNR	Peak Signal to Noise Ratio
QoS	Quality of Service
QP	Quantization Parameter
R-D	Rate-Distortion
RL	Reinforcement Learning
RLC	Radio Link Control
SM	Statistical Multiplexing
SNR	Signal to Noise Ratio
SVC	Scalable Video Coding
TC	Transform Coefficient
VBR	Variable Bit Rate
VOD	Video On Demand
VU	Video Unit

Contents

3

Remerciements **5**

Résumé **7**

Abstract **9**

Publications and patents **11**

List of acronyms **13**

Contents **15**

Résumé **19**

1 Introduction . . . 19
1.1 Motivation . . . 19
1.2 Problématique . . . 19
2 Partie I: Modèles . . . 20
3 Partie II: Multiplexage statistique des flux vidéo . . . 20
3.1 Multiplexage statistique sur un canal de diffusion . . . 20
3.2 Multiplexage statistique distribué . . . 25
4 Partie III: Transmission d'un flux scalable sur un réseau sans fil . . . 30
4.1 Introduction . . . 30
4.2 Adaptation de la tranmsission d'une vidéo scalable sur un réseau sans fil . . . 31
4.3 Apprentissage en ligne pour la transmission vidéo sur un réseau sans fil avec retour d'information retardé . . . 34
5 Conclusions . . . 38

1 Introduction **41**

1 Motivation . . . 41
2 Challenges . . . 41
3 Contributions . . . 42
3.1 Chapters 3&4: Statistical multiplexing over broadcast network 42
3.2 Chapters 6&7: Scalable video transmission over Unicast network 43

I Coding Tools for Video Rate Control **45**

2 Rate and Distortion Models for Video Rate Control **47**
1 Introduction . . . 47
2 Video encoders . . . 47
2.1 H.264/AVC . . . 48
2.2 H.264/SVC . . . 51
2.3 Rate control . . . 54
3 Rate Distortion models . . . 57
3.1 Independent models . . . 57
3.2 Dependent models . . . 60
4 Conclusion . . . 61

3 Independent Regime-Switching Auto-Regressive Model **63**
1 Introduction . . . 63
2 Experimental observations . . . 63
2.1 Rate regimes . . . 64
2.2 Distortion regimes . . . 65
3 Independent regime-switching auto-regressive model . . . 65
3.1 Main ideas . . . 65
3.2 Definition . . . 66
3.3 Rate and distortion characteristics . . . 67
3.4 Compound model . . . 68
3.5 Experimental results . . . 69
4 Conclusion . . . 71

II Statistical Multiplexing: Deterministic Optimization **73**

4 Video Statistical Multiplexing over Broadcast Networks **75**
1 Introduction . . . 75
1.1 Problem statement . . . 75
1.2 Related works . . . 76
1.3 Main contributions . . . 78
2 Architecture of the proposed statistical multiplexing system . . . 79
3 Optimization problem formulation . . . 81
3.1 Cost function . . . 82
3.2 Rate constraints . . . 82
3.3 Quality constraints . . . 87
3.4 Summarized constrained optimization problem . . . 88
3.5 Statistical multiplexing application in an MBMS system . . . 89
4 Experimental Results . . . 90
4.1 Constant channel rate . . . 91
4.2 Variable channel rate . . . 98
5 Conclusion and open issues . . . 105

5 Distributed Statistical Multiplexing **109**
1 Introduction . . . 109
1.1 Related works . . . 109
1.2 Main contributions . . . 110
2 Problem statement . . . 111
3 Proposed solution . . . 112
3.1 Centralized bandwidth allocation . . . 112
3.2 Distributed encoder control . . . 114
4 Equilibrium and stability analysis . . . 116
4.1 Source model . . . 116
4.2 Control of the buffer levels in bits . . . 116
4.3 Control of the buffer levels in seconds . . . 120
5 Control at the frame level . . . 124
6 Experimental results . . . 126
6.1 Gaussian sources . . . 126
6.2 Video sources . . . 134
7 Conclusion and open issues . . . 141

III Scalable Video Rate Control: Stochastic Optimization **143**

6 Introduction to Markov Decision Process for Video Delivery **145**
1 Introduction . . . 145
2 Markov Decision Process . . . 145
2.1 Definitions . . . 145
2.2 Example: Video encoding and transmission control . . . 147
3 Dynamic programming . . . 148
3.1 Policy iteration . . . 148
3.2 Value iteration . . . 148
3.3 Example: Solution . . . 149
4 Reinforcement Learning . . . 153
4.1 Definitions . . . 153
4.2 Learning the optimal policy . . . 153
5 Conclusion . . . 155

7 Cross-layer Optimization of Scalable Video Delivery over Wireless Channel **157**
1 Introduction . . . 157
1.1 Challenges . . . 157
1.2 Related works . . . 159
1.3 Main contributions . . . 161
2 Scalable layer filtering process for video delivery over a wireless channel 162
2.1 Video transmission system . . . 162
2.2 Markov decision process formulation . . . 163
2.3 Experimental results . . . 168
3 Conclusion . . . 174

8 On-line Learning for Scalable Video Filtering over Wireless Networks with Delayed Feedback **177**
1 Introduction . . . 177
1.1 Related works . . . 178
1.2 Main contributions . . . 179
2 System description . . . 180
2.1 Streaming server . . . 180
2.2 Proxy . . . 181
2.3 Base station and channel . . . 181
2.4 Receiver . . . 181
3 Model description . . . 182
3.1 MDP formulation . . . 182
3.2 System states . . . 182
3.3 Actions . . . 183
3.4 State transition probability matrix . . . 183
3.5 Reward function . . . 184
3.6 Learning the optimal policy . . . 186
4 Experimental results . . . 188
4.1 Simulation conditions . . . 188
4.2 Learning process . . . 191
4.3 Results: System performance . . . 191
5 Conclusion . . . 194

9 Conclusions and perspectives **197**

References **201**

Appendices

A Probability density function of E_j^0 **213**

B Stability of the proposed distributed SM systems **215**
1 Reminder about the stability of a linear system . . . 215
2 Control of the buffer level in bits . . . 216
3 Control of the buffer level in seconds . . . 219

C State transition probabilities **223**
1 System states . . . 223
1.1 Channel state . . . 223
1.2 Frame state . . . 223
1.3 Post-encoder buffer . . . 223
1.4 MAC buffer . . . 224
2 Model state: system state space reduction . . . 224
2.1 Frame state . . . 224
2.2 Buffer state . . . 225
2.3 Channel state . . . 226

Résumé

1 Introduction

1.1 Motivation

L'augmentation de la bande passante disponible dans les réseaux sans fil a permis le développement d'un grand nombre de services vidéo parmi lesquels, la télévision mobile, la vidéo à la demande, la visiophonie, etc. Même si une croissance significative des ressources sans fil est également attendue avec la disponibilité des réseaux 4G, les ressources requises pour transmettre des flux compressés vers des récepteurs mobiles à large capacité d'affichage, comme les téléphones intelligents, sont en croissance encore plus rapide. Ainsi, la demande croissante de programmes de qualité toujours meilleure exige une meilleure gestion de la bande passante disponible et de la manière dont les contenus sont compressés. Outre l'utilisation de la bande passante, la conception et la mise en oeuvre de manière efficace d'applications multimédia sensibles aux délais, à la qualité vidéo et à l'hétérogénéité des systèmes est difficile en raison des contraintes temps réel ainsi que les variations de l'environnement subies par le système, tel que les caractéristiques de la source, les besoins des utilisateurs, les conditions du canal, etc. Pour répondre à ces problèmes, un contrôle bout-en-bout de la transmission vidéo et de nouvelles techniques de gestion des ressources du réseau sont nécessaires.

1.2 Problématique

Dans les systèmes de diffusion (Broadcast/Multicast), les programmes vidéo sont codés ensuite transmis en parallèle sur un canal. Outre l'utilisation optimale du débit de canal disponible, la régulation bout-en-bout d'une chaine de diffusion de contenues vidéo peut permettre la satisfaction de plusieurs contraintes liées à la qualité ou le délai de transmission des programmes, par exemple, une qualité minimum, une équité en qualité, une variation lisse de la qualité au cours du temps, ainsi que des contraintes de délai. Trouver un système de régulation en mesure de satisfaire simultanément toutes ces contraintes dans le contexte de la diffusion vidéo est une tâche difficile. Ceci est principalement dû à la non-stationnarité des programmes vidéo. Les variations peuvent être dues, par exemple, à des changements de scène ou à une activité élevée dans un programme. Par ailleurs, la plupart des systèmes de régulation existants sont effectuées de manière centralisée où les serveurs se partagent en continu des informations sur la qualité vidéo de leurs programmes en vue de satisfaire une équité en qualité. Toutefois, lorsque les serveurs vidéo sont situés dans des endroits séparés, un système de régulation débit décentralisé est souhaitable.

En outre, la plupart des systèmes de contrôle et de gestion des ressources existants réalise une optimisation à court terme et ne s'intéressent qu'à la performance immédiate du système. Cependant, dans presque tous les problèmes de gestion des ressource, les décisions prises à un instant impactent non seulement la performance immédiates du système mais aussi les performances futures. Faire un contrôle avec une commande prédictive exploitant la dynamique du système ou son estimée, peut améliorer l'utilisation des ressources, les délais de transmission et la qualité des programmes vidéo reçus. Cette thèse aborde ces problèmes et comporte trois parties

- La partie I rappelle les outils techniques de codage vidéo nécessaires pour la résolution des systèmes de contrôle proposés.
- La partie II présente un système de multiplexage statistique pour un ensemble de programmes vidéo transmis sur un canal commun. Ce problème est formulé comme un problème d'optimisation sous contraintes et résolu avec les outils d'optimisation déterministe.
- La partie III propose un processus de filtrage de couches pour la transmission de vidéos scalables sur un réseau type sans fil visant à maximiser la qualité vidéo à la réception. Ce problème est formulé comme un problème de d'optimisation stochastique. Il est résolu à l'aide des outils de programmation dynamique et d'apprentissage.

2 Partie I: Modèles

Après avoir introduit les conceptes de base du codage vidéo dans le chapitre 2, nous proposons une analyse des différents modèles débit distorsion existants. Ces outils sont fondamentaux pour les systèmes de contrôle du débit permettant le réglage à la fois du débit et de la qualité vidéo du flux codé. Dans le chapitre 3, nous nous sommes intéressés à la modélisation de la dépendance inter-images. La dépendance est spécifiée en termes d'impact des paramètres de codage d'une image sur le débit et la distorsion de l'image suivante. Cette dépendance est illustrée expérimentalement puis interprétée théoriquement en considérant la quantification d'un modèle à changements de régime autorégressif avec des paramètres de quantification variables. Les résultats expérimentaux montrent qu'en utilisant le modèle dépendant proposé, une représentation précise de la dépendance entre les images en distorsion est obtenue. Pour le débit, l'ajustement avec la partie texture de ce débit est satisfaisant, sauf pour les grandes valeurs de paramètre de quantification.

3 Partie II: Multiplexage statistique des flux vidéo

3.1 Multiplexage statistique sur un canal de diffusion

Dans le chapitre 4, nous proposons un système de multiplexage statistique (MS) des contenus vidéo sur un canal de diffusion. Le MS des contenus vidéo vise à transmettre plusieurs flux vidéo codés en débit variable sur un canal à bande limitée. Stocker les flux codés dans des mémoires tampon à la sortie des codeurs est l'une des diverses techniques utilisées pour lisser les fluctuations du débit de codage en raison

de variations dans l'activité des contenus vidéo. Outre que l'utilisation optimale du débit canal disponible, le système de MS à pour objectif la satisfaction de plusieurs contraintes, y compris la qualité de la vidéo et le délai de transmission. Pour cela nous avons ciblé les contraintes suivantes

- *remplissage du canal*: le débit canal doit être efficacement utilisé,
- *qualité minimum* : les programmes sont codés avec une qualité minimum,
- *équité* en qualité : les programmes sont codés avec une qualité similaire,
- *variation* de la qualité : les variations de la qualité de chaque programme décodé au cours du temps sont faibles,
- *niveau des tampons*: les niveaux en bits des tampons sont contrôlés,
- *délai*: les délais de transmission sont minimisés.

Pour satisfaire ces contraintes, nous proposons un contrôle conjoint en boucle fermée des codeurs vidéo et des tampons qui permet de contrôler simultanément les paramètres de codage de tous les codeurs vidéo ainsi que les débits de transmission pour chaque tampon. Les paramètres de tous les codeurs vidéo sont ajustés pour chaque Unité Vidéo (UV). Une UV peut être une image ou un groupe d'images. Le contrôle des paramètres de codage se fait à l'aide d'une commande prédictive sur une fenêtre contenant l'UV précédente, l'UV actuelle, et $W-2$ UVs futures. Cette technique permet une meilleure satisfaction des contraintes de qualité par rapport au contrôle non prédictif. Ceci nécessite d'estimer les caractéristiques débit-distorsion de chaque UV impliqué dans la fenêtre de contrôle.

La gestion des tampons est abordée de deux façons différentes: d'abord, les niveaux en bits de tous les tampons sont contrôlés simultanément afin de prévenir d'avoir des tampons trop plein ou trop vide; deuxièmement, un délai de transmission similaire pour tous les programmes est recherché.

Les niveaux des tampons sont ajustés via le débit de transmission de chaque programme pour utiliser pleinement le débit canal disponible et contrôler le niveau en bits ou en secondes des tampons. La boucle fermée est obtenue en utilisant un contrôle Proportionnel-Intégral-Dérivé (PID).

L'idée principale est de mettre à jour dynamiquement le débit de codage pour chaque UV en fonction du niveau moyen des tampons envoyé au contrôleur des codeurs vidéo, afin de maximiser la qualité de chaque programme et d'utiliser efficacement la bande disponible.

3.1.1 Architecture du système de multiplexage statistique proposé

La figure 1 présente l'architecture pour le MS de N vidéo codées et transmises en parallèle sur un canal de diffusion. Dans ce contexte, aucun retour d'information du récepteur n'est considéré.

Le processus de contrôle des codeurs délivre un paramètre de quantification pour chaque UV pour chacun des N codeurs vidéo.

A chaque instant j, le contrôleur des codeurs détermine le paramètre de quantification Q_{ij} pour le i ème codeur de sorte que les débits de codage R^{e}_{ij}, pour tout

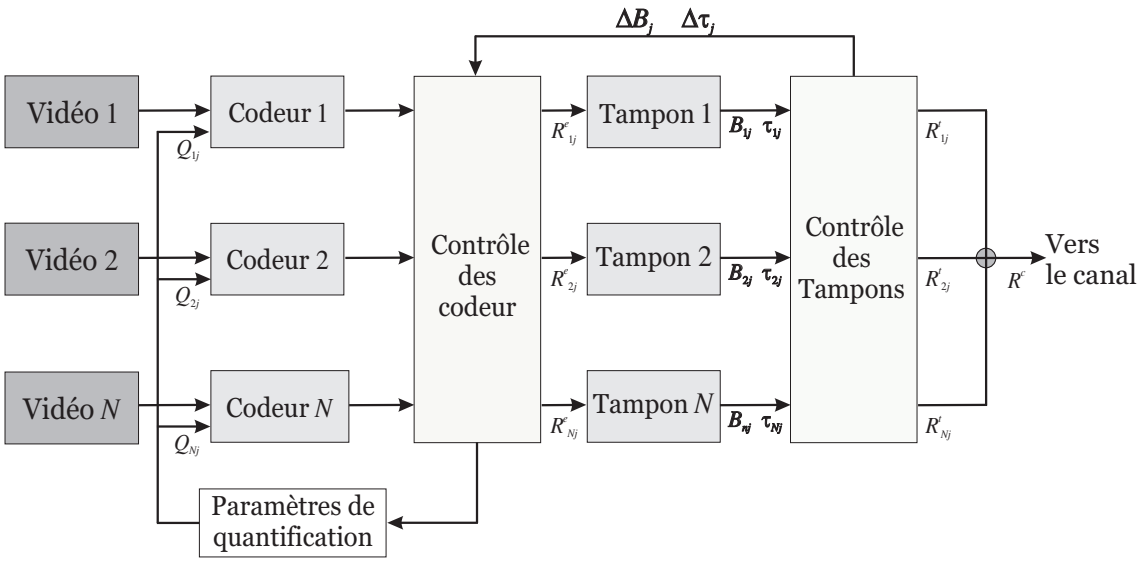


Figure 1: Contrôle en boucle fermée du systéme de multiplexage statistique proposé.

$i = 1 \dots N$ satisfassent la contrainte de débit de codage tout en satisfaisant les contraintes de qualité.

A chaque sortie de codeur, un tampon stocke temporairement les UV codées. Le processus de contrôle des tampons permet de réguler le niveau de remplissage des tampons au niveau bit B_{ij} ou au niveau délai τ_{ij} en utilisant des niveaux de références noté B_0 (en bits) et τ_0 (en secondes). Cette régulation se fait à partir des débits de transmission R^{t}_{ij} alloués à chaque programme $i \in \{1 \dots n\}$ de sorte que l'utilisation de la bande soit optimale.

Les différences $B_{ij} - B_0$ et $\tau_{ij} - \tau_0$ moyennées sur les N programmes sont notées ΔB_j et $\Delta\tau_j$. ΔB_j et $\Delta\tau_j$ sont envoyés au contrôleur des codeurs qui les utilisent pour mettre à jour la contrainte de débit R_{j+1} pour la prochaine UV. Les codeurs vidéo et les tampons sont ainsi contrôlés en boucle fermée, voir figure 1.

3.1.2 Problème d'optimisation

L'objectif du système de MS proposé est de fournir à chaque instant j des paramètres de quantification Q_{ij} aux codeurs vidéo et le débit de transmission R^{t}_{ij} pour les tampons de façon à satisfaire la liste des contraintes considérées. En raison des variations des caractéristiques débit-distorsion des contenus vidéo, certaines décisions immédiates prises à l'instant j, en considérant uniquement l'état du système à ce même instant, peuvent conduire à des violations des contraintes aux instant futurs $j' > j$.

La solution proposée pour ce problème est d'effectuer le contrôle des codeurs vidéo sur une fenêtre temporelle de W UVs pour chaque programme, de UV $j-1$ à UV $j+W-2$. A l'instant j et pour chaque programme i, le processus de contrôle des codeurs évalue un vecteur $\mathbf{Q}_i^{(j)} = \left(Q_{ij}^{(j)} \dots Q_{ij+W-2}^{(j)}\right)$.

Seuls les paramètres $Q_{ij}^{(j)}$ évalués pour la j-ème UV sont appliqués à l'instant j aux codeurs $i \in \{1 \dots N\}$, les paramètres $Q_{ij+k}^{(j)}$ évalués pour les UVs futures, $k = 1 \dots W-2$, ne sont pas appliqués, mais mis à jour au cours des prochaines étapes. Ce contrôle permet de prévoir le choix des paramètres de contrôle $\mathbf{Q}_i^{(j)}$ qui

satisfait les contraintes à l'instant j et pour lesquels on sait qu'il existe des valeurs $Q_{ij+k}^{(j)}$ tels que les contraintes seront aussi satisfaites aux instants futures pris en compte dans la fenêtre de contrôle.

Le problème est ensuite formulé sous forme de problème d'optimisation sous contrainte. Les paramètres de quantification sont choisis pour chaque flux vidéo de façon à ce que la qualité vidéo correspondante moyennée sur les N programmes diffusés soit maximisée sous contraintes de qualité vidéo et de bande. Une fois codé, chaque flux est temporairement ensuite stocké dans son tampon respectif. A ce niveau, un deuxième système de contrôle est déployé pour obtenir les débits de transmission pour chaque programme de façon à ce que le débit canal soit entièrement utilisé et les niveaux des tampons soient contrôlés. A l'issue de ce deuxième contrôle, un nouveau niveau de tampon est obtenu. L'information sur le niveau de remplissage des tampons est alors envoyée vers le contrôleur des codeurs vidéo pour mettre à jour la contrainte de débit de codage. Ainsi, lorsque le niveau moyen des tampons est plus élevé que le niveau de remplissage de référence, le débit de codage de l'UV suivante devrait être réduit et inversement. Cette mise à jour de la contrainte de débit de codage est réalisée à l'aide d'un contrôleur PID.

3.1.3 Etude expérimentale

Nous avons évalué la performance du système de MS proposé en utilisant quatre programmes vidéo codés avec l'codeur H.264/AVC, multiplexés et transmis sur un canal de diffusion. Chaque programme vidéo est constitué de plusieurs séquences différentes. Cela permet de simuler les changements de scène qui peuvent se produire dans un programme vidéo. Le débit canal est d'abord considéré constant et égal à 1 Mbits/s. Les deux scenarios de contrôle des tampons sont considérés: contrôle en bits et contrôle en secondes. La commande prédictive est réalisée en utilisant une fenêtre de contrôle de $W = 4$ et est comparé à un scénario de référence sans commande prédictive ($W = 2$), mais pour lequel la contrainte de variation de la qualité est imposée.

Dans les deux cas, la somme des débits de transmission est égale au débit canal grâce au contrôle des tampons que ce soit au niveau bits ou au niveau secondes. Grace à la contrainte de l'équité en qualité, une qualité similaire est obtenue pour tous les programmes transmis.

La contrainte de variation de la qualité introduite réduit l'amplitude des variations du PSNR, voir figure 2, où l'écart-type du PSNR est représenté pour plusieurs valeurs de débit de canal sans contrainte de variation de qualité et avec contrainte de variation de qualité lorsque $W = 2$ et $W = 4$ et lorsque le niveau en secondes des tampons est contrôlé.

Le contrôle en boucle fermée a permis aux quatre tampons d'avoir des niveaux de remplissage (bits ou secondes) similaires et autour du niveau de référence imposé (B_0 ou τ_0). De plus, pour $W = 4$, la variation des niveaux des tampons est plus lisse que pour $W = 2$. Ceci est dû à la commande prédictive du débit de codage effectuée afin de mieux satisfaire les contraintes de qualité.

Le tableau 1 montre la performance du système en termes de l'écart entre le delai et la valeur référence τ_0, noté $\Delta\tau$, et la variance du délai, noté σ_τ^2, avec et sans commande prédictive en utilisant un contrôleur P, PI, ou PID. On voit que le contrôle PI réduit l'écart de délai $\Delta\tau$. Le terme dérivé pour cet exemple réduit

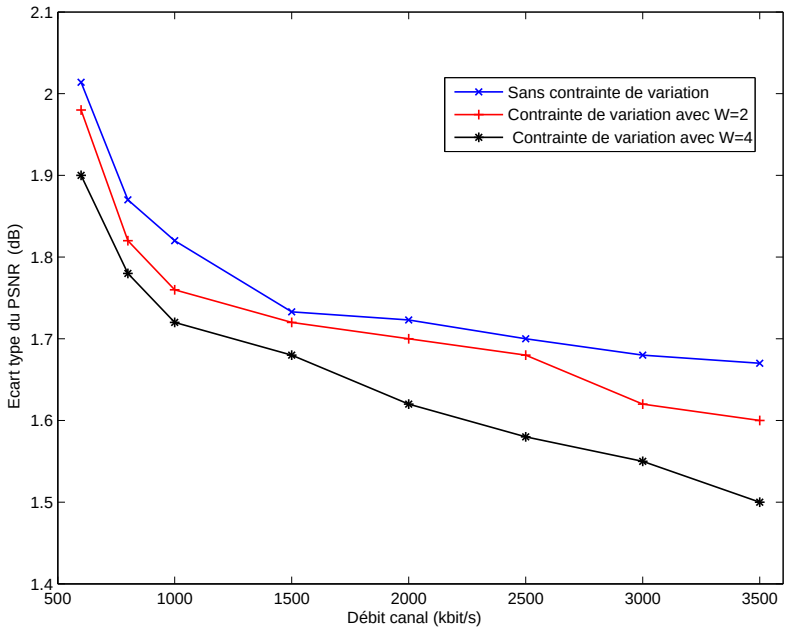


Figure 2: Ecart-type du PSNR sans contrainte de variation de qualité et avec contrainte de variation de qualité lorsque $W = 2$ et $W = 4$ pour différentes valeurs de débit de canal.

K_P, K_I, K_D	$W = 2$		K_P, K_I, K_D	$W = 4$	
	$\Delta\tau$	σ_τ^2		$\Delta\tau$	σ_τ^2
$0.2, 0, 0$	0.02	0.05	$0.2, 0, 0$	0.01	0.013
$0.2, 0.01, 0$	0.01	0.04	$0.2, 0.01, 0$	0.003	0.016
$0.2, 0.01, 0.01$	0.01	0.035	$0.2, 0.01, 0.05$	0.003	0.015

Table 1: Performance du système en terme de $\Delta\tau$ et σ_τ^2 en utilisant un contrôleur P, PI, ou PID avec $W = 2$ et $W = 4$ utilisant un débit de canal constant et le contrôle des tampons en seconde.

légèrement la variance délai σ_τ^2.

Nous avons également évalué la robustesse de notre système de contrôle à l'égard des variations du débit de canal. Pour cela, nous avons testé notre système de contrôle avec un débit de canal variable au cours du temps. En fait, ces variations peuvent être dû à des services concurrents, ce qui peut laisser des ressources plus ou moins au service de la diffusion. Les mêmes performances en termes de contrainte de bande, de qualité vidéo et de tampon sont obtenues dans ce cas de simulation. Ces résultats confirment que le système de contrôle proposé est robuste aux variations des caractéristiques du contenu vidéo et aussi au débit de canal.

3.1.4 Conclusion

Une commande prédictive pour un système de MS utilisant le codeur vidéo H.264/AVC a été proposée dans le cadre de la diffusion vidéo. Le système proposé effectue un contrôle en boucle fermée des codeurs et des tampons en utilisant un contrôleur PID. Le contrôle vise une utilisation efficace du débit de canal disponible tout en satisfaisant une qualité vidéo minimale, une équité en qualité entre les programmes,

et de faible variations de qualité au cours du temps. Un faible délai de transmission est également recherché.

La performance du système proposé a été évaluée par des simulations. Les résultats expérimentaux montrent qu'avec un débit de canal constant ou variable, grâce à la commande prédictive et au contrôle en boucle fermée des codeurs et des tampons, le canal est efficacement utilisé, les contraintes de qualité vidéo sont satisfaites ainsi que les contraintes sur les tampons.

3.2 Multiplexage statistique distribué

Dans le chapitre 4, le système de MS proposé est effectué de façon centralisée. Les programmes multiplexés doivent partager des informations sur la qualité de leur continu vidéo afin de satisfaire la contrainte de l'équité en qualité.

Dans le chapitre 5, nous proposons un système de MS partiellement distribué en mesure de satisfaire une contrainte d'équité en qualité vidéo entre les programmes sans échanger des informations entre les serveurs. L'allocation de bande passante entre les programmes est centralisée et faite dans un élément du réseau pouvant avoir accés à des informations sur le contenu multimédia transmis. Ceci permet de tenir compte de la contrainte d'équité en qualité. Cet élément du réseau renvoie à chaque serveur vidéo le niveau du tampon auquel il est associé. Ainsi, chaque serveur vidéo ajuste son débit de codage indépendamment des autres pour que son tampon atteigne un certain niveau de référence en bits ou en secondes. Le problème de MS est représenté par un système de contrôle avec retour d'information. Un contrôleur PI pour le contrôle de la bande passante est utilisé pour ajuster le débit de transmission à extraire de chaque tampon en fonction de la qualité vidéo moyenne du flux codé qu'il contient. Un deuxième contrôleur PI pour le contrôle du débit de codage est utilisé pour ajuster le débit de codage pour chaque codeur vidéo en fonction du niveau de leur tampon respectif en bits ou en secondes. L'étude de la stabilité du système est menée et la région de stabilité du système est caractérisée.

3.2.1 Enoncé du problème

Nous considérons un système de diffusion dans lequel N programmes vidéo sont codés et transmis en parallèle sur un canal dont le débit de transmission est R^{c}, voir figure 3. Le contrôle se fait pour chaque UV. A l'instant j, les j-ème UVs codées sont envoyées par les codeurs vidéo. Pour chaque flux vidéo $i \in \{1 \dots N\}$ un contrôle du codage individuel est effectué pour chaque UV générant un débit de codage R^{e}_{ij} et un PSNR P_{ij} pour chaque vidéo $i \in \{1 \dots N\}$. Chaque UV codée est alors transmise sur un réseau filaire et stockée dans son tampon correspondant dans l'élément du réseau sensible au contenu multimédia. Le débit de canal disponible est ensuite réparti entre les N programmes de telle manière que tous les programmes ont une qualité comparable.

3.2.2 Solution proposée

Nous proposons une solution semi-distribuée pour le problème proposé. L'idée est de réguler le flux vidéo en utilisant deux contrôleurs. Le premier contrôleur effectue *un contrôle du codage* individuel pour chaque codeur vidéo. Le second contrôleur

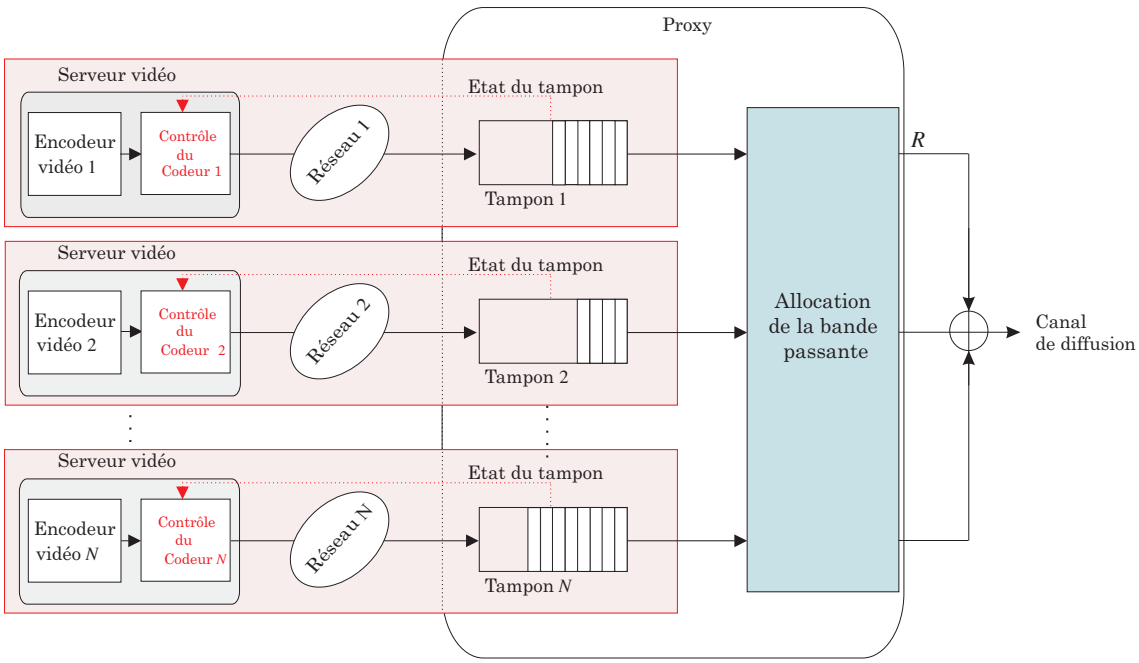


Figure 3: Système de multiplexage statistique des flux vidéo fournis par des serveurs distants.

effectue un contrôle centralisé de *l'allocation de bande passante* ciblant une qualité vidéo comparable entre les N programmes en utilisant les informations PSNR des UV déjà stockées dans les tampons. Les deux processus de contrôle sont effectués en boucle fermée. L'information du niveau de remplissage de chaque tampon est renvoyé au serveur vidéo correspondant. Ce dernier l'utilise pour adapter le débit de codage de la prochaine UV afin que le tampon atteigne un certain niveau de référence en bits ou en seconde.

Contrôle de l'allocation de bande passante

A chaque instant j, le débit de canal disponible R^{c} est réparti entre les programmes vidéo de manière centralisée de façon à ce que la contrainte de l'équité en qualité soit satisfaite. Un contrôleur PI est utilisé pour calculer le débit de transmission R^{t}_{ij} entre l'instant $j-1$ et l'instant j utilisant un débit de référence $R_0 = \frac{R^{\mathrm{c}}}{N}$. L'équation de contrôle du débit de transmission avec un contrôleur PI est

$$R^{\mathrm{t}}_j = R_0 + K^{\theta}_{\mathrm{P}}(\bar{P}_{j-1} - P_{ij-1}) + K^{\theta}_{\mathrm{I}} \sum_{k=1}^{j-1} (\bar{P}_k - P_{ik}), \qquad (1)$$

tel que K^{θ}_{P} est le gain proportionnel et K^{θ}_{I} est le gain proportionnel-intégral de correction. On suppose que chaque serveur vidéo i fournit à l'élément du réseau le PSNR P_{ij-1} de la $j-1$-ème UV et que le PSNR moyen $\bar{P}_{j-1}$ entre les N programmes peut ainsi être calculé.

Contrôle du codage

Pour chaque codeur vidéo un processus de contrôle du codage individuel est effectué au niveau de chaque UV. Le débit de codage peut être ajusté selon le niveau du tampon en bits ou en secondes en fonction des applications et des exigences du système. Le contrôle du codage est réalisé à l'aide d'un contrôleur PI. Dans le cas

de la régulation des niveaux des tampons en bits, l'équation de contrôle de débit de codage selon le niveau du tampon en bits est

$$R_{ij}^{\mathrm{e}} = R^0 - K_{\mathrm{P}}^{\mathrm{bR}} \left(\frac{B_{ij-1} - B_0}{T} \right) - K_{\mathrm{I}}^{\mathrm{bR}} \sum_{k=1}^{j-1} \left(\frac{B_{ik} - B_0}{T} \right), \tag{2}$$

tel que $K_{\mathrm{P}}^{\mathrm{bR}}$ et $K_{\mathrm{I}}^{\mathrm{bR}}$ sont les gains proportionnel et proportionnel-intégral de correction. On voit à partir de (2) que le débit de codage augmente lorsque le niveau du tampon est inférieur au niveau de référence B_0. Puisque le niveau du tampon dépend également du débit de transmission alloué qui lui dépend du PSNR moyen $\bar{P}_{j-1}$. Au final le débit de codage sera augmenté ou diminué, selon le PSNR des UVs déjà codées.

3.2.3 Analyse de la stabilité du système

Nous avons étudié la stabilité du système de contrôle de l'allocation de bande passante et du débit de codage utilisant un contrôleur PI. Le système de contrôle en boucle fermée doit être stable pour répondre à l'objectif des performances requises.

L'étude de la stabilité du système de contrôle proposé, lorsque le niveau des tampons en bits est régulé, se fait par l'étude des équations d'état à partir desquelles la région de stabilité est caractérisée. Quand le contrôle du codage se fait en fonction du niveau du tampon en secondes, les équations d'état, non-linéaires, ont été linéarisées autour de l'équilibre pour caractériser la région de stabilité du système dans ce cas de contrôle. Cette région de stabilité permet de définir les valeurs des paramètres de gain de correction des contrôleurs à implémenter pour que le système soit stable.

3.2.4 Etude expérimentale

Le système de contrôle proposé peut s'appliquer à des codeurs vidéo AVC ou SVC. Nous avons évalué notre système de contrôle en utilisant quatre fournisseurs de contenus vidéo diffusant des vidéos différentes codées avec des codeurs tel que H.264/AVC. Le contrôle se fait au niveau de chaque groupe d'images.

Les paramètres du contrôle de l'allocation de la bande passante et du contôle de codage utilisant deux contrôleurs PI sont fixés à partir de la région de stabilité caractérisée.

Les niveaux de remplissage des tampons en bits, les PSNRs, et les délais en seconde correspondant à chaque programme sont représentés sur la figure 4 lorsque les niveaux des tampons en bits sont contrôlés, et sur la figure 5, lorsque les niveaux des tampons en secondes sont contrôlés.

On peut voir que, dans les deux cas, les contrôleurs PI assurent dans la plupart du temps une déviation des niveaux des tampons par rapport aux niveaux de référence faible. Suite à des changements de scène, on peut voir que le système de contrôle atteint l'équilibre après une courte phase transitoire pour satisfaire la contrainte de l'équité en qualité.

La régulation du niveau du tampon en secondes, permet à tous les programmes multiplexés d'avoir plus ou moins le même délai autour de la même valeur de référence τ_0. Dans ce cas, la performance du système en termes d'écart de délai Δ_τ, la variance du délai σ_τ^2, l'écart moyen entre le PSNR et le PSNR moyen Δ_{PSNR},

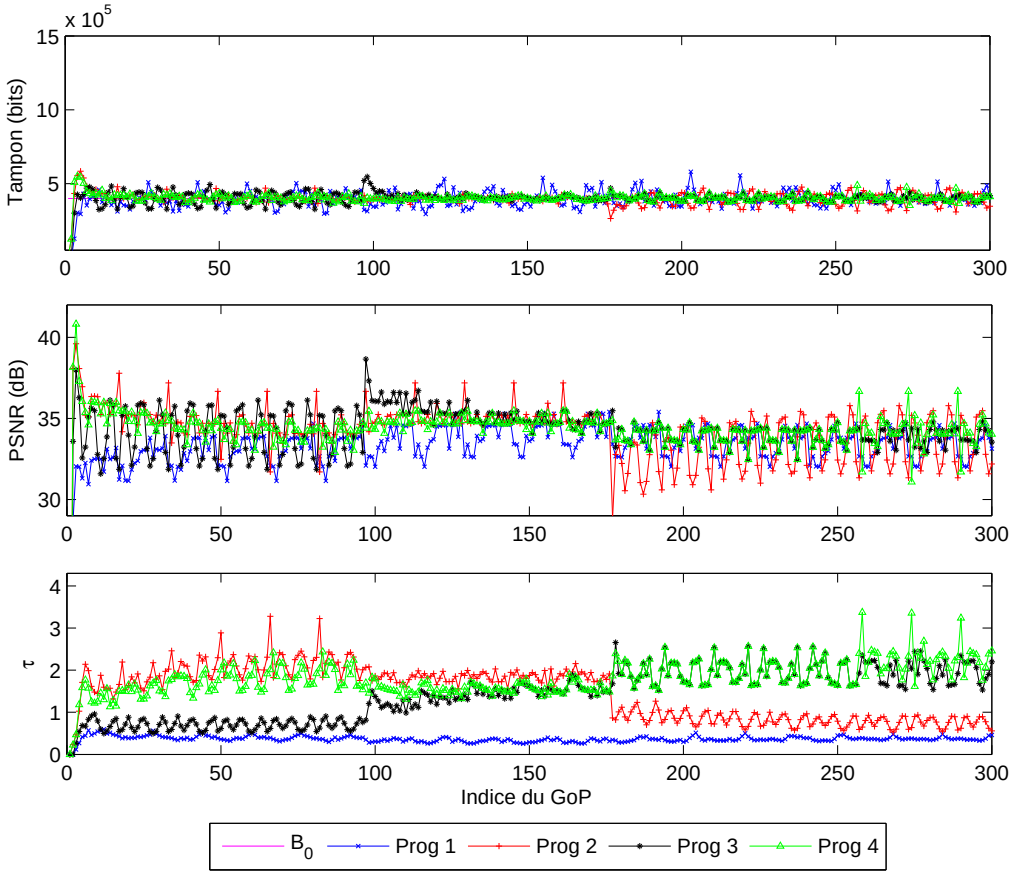


Figure 4: Performance du système en utilisant un contrôleur PI pour le contrôle de la bande passante et un contrôleur PI pour le contrôle du codage fait à partir du niveau des tampons en bits en multiplexant quatre programmes vidéo.

$K_{\mathrm{P}}^{\theta}, K_{\mathrm{I}}^{\theta}$	$K_{\mathrm{P}}^{\tau\mathrm{R}}, K_{\mathrm{I}}^{\tau\mathrm{R}}$	Δ_{τ}	σ_{τ}^2	Δ_{PSNR}	σ_{PSNR}^2
$2.10^4, 0$	$2.5.10^4, 0$	2	0.39	6.25	3.9
$2.10^4, 0$	$2.5.10^4, 5.10^3$	0.08	0.32	6.33	3.85
$2.10^4, 1.10^3$	$2.5.10^4, 5.10^3$	0.21	0.45	5.99	1.5

Table 2: Performance du système en termes de Δ_{τ}, σ_{τ}^2, Δ_{PSNR}, et σ_{PSNR}^2, quand des contrôleurs P et PI sont utilisés pour le contrôle de la bande passante et pour le contrôle des codeurs fait à partir du niveau des tampons en secondes en multiplexant quatre programmes vidéo.

et la variance moyenne des PSNRs σ_{PSNR}^2 sont données dans le tableau 2, en utilisant des contrôleurs P et PI avec

$$\sigma_{PSNR}^2 = \frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{M}\sum_{l=1}^{M}\left(P_{nl}-\bar{P}_l\right)^2\right), \tag{3}$$

$$\Delta_{PSNR} = \left|\frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{M}\sum_{l=1}^{M}\left(P_{nl}-\bar{P}_l\right)\right)\right|, \tag{4}$$

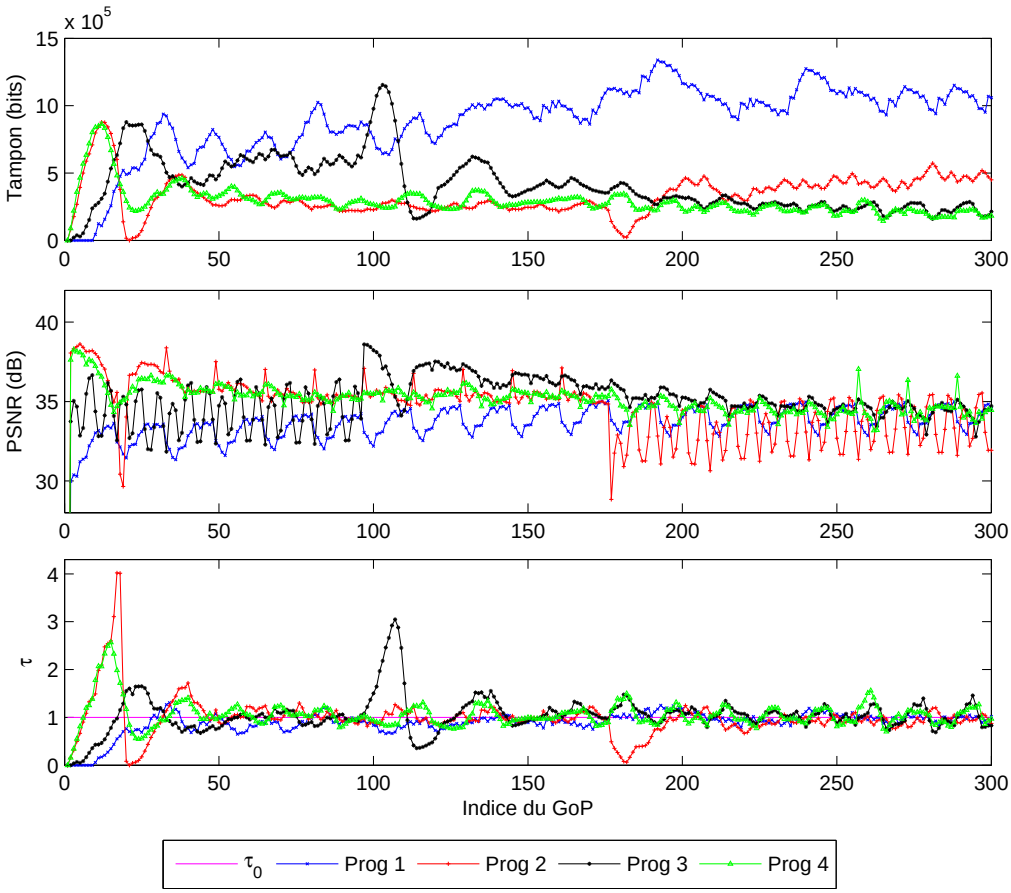


Figure 5: Performance du système en utilisant un contrôleur PI pour le contrôle de la bande passante et un contrôleur PI pour le contrôle du codage fait à partir du niveau des tampons en secondes en multiplexant quatre programmes vidéo.

$$\Delta_\tau = \left| \frac{1}{N} \sum_{n=1}^{N} \left(\frac{1}{M} \sum_{l=1}^{M} (\tau_{nl} - \tau_0) \right), \right| , \tag{5}$$

et

$$\sigma_\tau^2 = \frac{1}{N} \sum_{n=1}^{N} \left(\frac{1}{M} \sum_{l=1}^{M} (\tau_{nl} - \tau_0)^2 \right). \tag{6}$$

En régulant les niveaux des tampons en bits ou en secondes, le contrôleur PI réduit l'erreur entre le niveau du tampon et le niveau de référence par rapport au contrôleur P sans perturber le comportement du contrôle de l'allocation de bande puisque la variance du PSNR reste la même.

3.2.5 Conclusion

Nous avons proposé un système de contrôle qui permet le MS de plusieurs sources vidéo de manière semi-distribuée. Deux contrôleurs sont impliqués dans le système de MS. Le premier, centralisé, est en charge d'allouer la bande passante entre les

programmes afin de satisfaire une contrainte d'équité en qualité. Le deuxième, décentralisé, se fait au niveau des codeurs vidéo utilisant l'information sur le niveau de remplissage du tampon correspondant pour contrôler le débit de codage de sorte qu'un niveau de tampon référence est atteint. Le processus de contrôle implique deux contrôleurs PI. L'étude de stabilité a permis de caractériser la région de stabilité du système permettant de définir les paramètres des contrôleurs P et PI à considérer pour que le système soit stable. Les tests expérimentaux montrent que la contrainte d'équité en qualité est satisfaite et que le système est stable.

4 Partie III: Transmission d'un flux scalable sur un réseau sans fil

Dans cette partie, on s'intéresse au problème de transmission des vidéos scalables sur un canal sans fil. L'objectif est de maximiser la qualité vidéo à la réception. Pour cela nous proposons dans les chapitres 7 et 8 deux systèmes de filtrages de couches qui permettent de contrôler dans le réseau l'envoi des couches scalables de façon à contrôler conjointement les niveaux des tampons dans le réseau et la qualité vidéo reçue. Le premier schéma de contrôle proposé dans le chapitre 7 permet un contrôle à la fois des tampons au niveau de l'émetteur et du récepteur. Le problème est représenté et résolu avec le Processus de Décision Markovien (PDM) qui définit pour chaque état du système la meilleure décision à prendre en termes de filtrage de couches. Le PDM permet la conception de décision prédictive qui maximise une récompense à long terme. L'évaluation de cette décision se fait hors ligne en supposant que le contrôleur connait la dynamique de variation de l'environnement qui l'entoure. Dans le chapitre 8, un schéma de contrôle en ligne est proposé. Un apprentissage par renforcement est considéré pour apprendre en ligne les décisions à prendre dans chaque état. Ce système de filtrage traite également le problème de retard entre la prise de décision et la disponibilité des états du canal.

4.1 Introduction

Un processus de décision Markovien (PDM) est défini comme un processus stochastique contrôlé satisfaisant la propriété de Markov, assignant des récompenses aux transitions d'états. Le modèle PDM peut être vu comme une chaîne de Markov à laquelle on ajoute une composante décisionnelle. On les décrit par un quintuplet : $(\mathcal{S}, \mathcal{A}, P, r)$:

- $\mathcal{S}$ est l'espace d'états dans lequel évolue le processus,
- $\mathcal{A}$ est l'espace des actions qui influent sur la dynamique de l'état,
- P sont les probabilités de transition entre états,
- r est la fonction de récompense sur les transitions entre les états.

Le PDM est bien adapté pour les problèmes d'optimisation d'un système qui satisfait la propriété de Markov. L'optimisation de la transmission vidéo est l'un des nombreux exemples qui peuvent être abordés avec cette méthode en raison de

la déscription Markovienne que l'on peut faire du comportement du contenu vidéo, de l'état des tampons, et du canal de transmission.

A l'aide de technique de programmation dynamique, il est possible de déterminer la commande optimale

- lorsque l'état du système est parfaitement connu,
- en présence d'incertitude sur l'effet des actions.

Dans les problèmes où les fonctions de transition P et de récompense r sont inconnues, les méthodes d'apprentissage par renforcement proposent d'apprendre la décision optimale à prendre dans chaque états. À chaque pas de temps de l'algorithme, l'agent perçoit son état et dispose d'un ensemble d'ations possibles. Il choisit une action et reçoit du système un nouvel état et une récompense. A l'aide de ces informations, l'algorithme d'apprentissage par renforcement doit permettre à l'agent de développer une politique qui lui permet de maximiser la récompense à long terme.

4.2 Adaptation de la tranmsission d'une vidéo scalable sur un réseau sans fil

On s'intéresse essentiellement à l'optimisation de la qualité vidéo reçue lors d'une transmission vidéo en unicast. Pour maximiser la qualité vidéo, plusieurs éléments du réseau peuvent être contrôlés, y compris, la stratégie d'ordonnancement, les tampons dans chaine de transmission, le taux de perte, etc. Nous proposons un système de contrôle au coeur du réseau qui effectue un filtrage entre les différentes couches d'un flux scalable codé et stocké dans des tampons de post-encodage. Ce système effectue un filtrage des couches avec prise en compte de l'ordre de priorité entre les couches. Ce contrôle a pour but de maximiser la qualité de la vidéo à la réception.

4.2.1 Description du problème

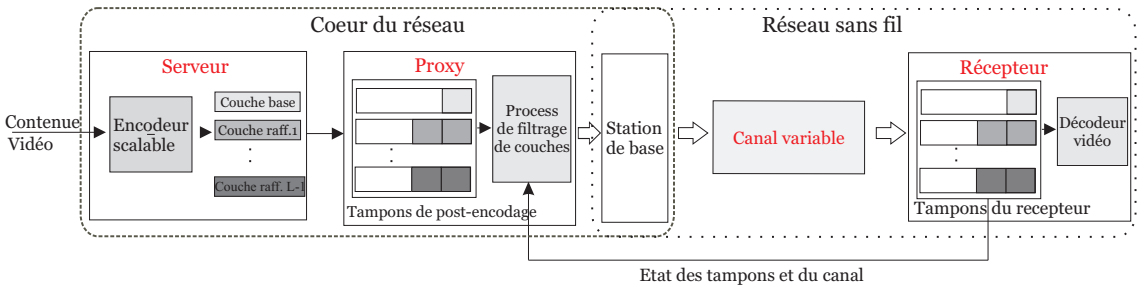


Figure 6: Schéma de transmission vidéo bout-en-bout.

Dans un système de transmission vidéo vers un récepteur mobile, on identifie plusieurs éléments tel que la source vidéo, le codeur de source, les tampons répartis sur plusieurs endroits du réseau, un canal sans fil, et le récepteur, voir figure 6. Selon la localisation du système de contrôle de transmission vidéo, certains de ces éléments peuvent être contrôlables et d'autres pas. Dans notre cas, le système de

contrôle est placé dans un proxy dans le coeur du réseau. Il est en charge de filtrer les couches scalables stockées dans des tampons dans le *proxy*. Ce filtrage exploite la propriété de séparabilité des couches scalables pour adapter le flux codé sans avoir à ré-encoder la séquence. Ce filtrage se fait en se basant sur des critère de qualité d'expérience qui ciblent une qualité vidéo maximale à la réception. Ces critères sont le PSNR et une marge minimale dans le tampon du récepteur qui évite le gel de la vidéo au décodage lors de dégradation des conditions du canal.

Le processus de filtrage proposé doit être conçu pour que les conditions suivantes soient satisfaites:

1. La qualité vidéo décodée doit être maximisée,
2. Le débit canal disponible doit être utilisé efficacement,
3. Les niveaux des tampons de post-encodage doivent être ajustés afin d'éviter tout débordement,
4. Les tampons au niveau du récepteur doivent être contrôlés afin de maintenir un niveau de remplissage minimal afin d'être robuste à l'égard des variations du canal.

Le problème d'optimisation est formulé dans le cadre du processus de décision Markovien. Toutes les contraintes citées doivent être traduites dans la fonction récompense. Pour cela nous avons définit les états systèmes composé de l'état h_t de la station de base et le canal, les états s^e_ℓ, $\ell = 1, \ldots, L$ des tampons de post-encodage des L couches SVC placés dans le proxy, et les états s^r_ℓ des tampons du récepteur.

A chaque instant t, échantillonné à la période d'une image, le proxy doit prendre une action a_t qui détermine le nombre de couches scalables à envoyer. Lorsque les conditions du canal sont mauvaises, pour éviter le débordement des tampons de post-encodage, les paquets peuvent également être détruits. Quand les tampons de post-encodage l'autorisent, le proxy peut également choisir de ne rien envoyer et d'attendre jusqu'à ce que les conditions du canal s'améliorent.

4.2.2 Résolution du problème

Le proxy choisi l'action qui maximise la qualité vidéo reçue et qui permet également de satisfaire les contraintes de débit de canal, de niveau de remplissage des tampons de post-encodage et du récepteur. Ainsi, une fonction de récompense r qui prend en compte toutes ces contraintes est considérées

$$r_t(\mathbf{s}_t, \mathbf{a}_t) = \underbrace{\sum_{l=1}^{L} \gamma_l a_{l,t}}_{\text{Rec. de transmission}} + \underbrace{\beta\nu \left(R_t^T(\mathbf{a}_t, h_t) - R_t^{\mathrm{c}}\right)}_{\text{Rec. de de débit canal}} + E\left[\underbrace{\sum_{l=1}^{L} \lambda_l \rho(s^e_{l,t}, a_{l,t})}_{\text{Rec. des tampons de post-encodage}} + \underbrace{\sum_{l=1}^{L} \mu_l \rho(s^r_{l,t}, a_{l,t})}_{\text{Rec. des tampons du recepteur}} \right]. \tag{7}$$

Les paramètres positifs γ_l, λ_l, μ_l, avec $l = 1 \dots L$, et β permettent de définir l'importance relative des différentes contraintes imposées au système. La fonction récompense dans (7) implique plusieurs parties chacune correspond à une des contraintes exigées.

Contrairement aux systèmes de contrôle de débit traditionnels, qui se concentrent sur le contrôle à court terme, où seule la fonction récompense immédiate est maximisée, l'objectif du contrôle proposé est de trouver les actions qui maximisent la somme amortie des espérances des récompenses futures

$$V^{\pi} = E_{\pi}[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t], \tag{8}$$

où le paramètre $0 \leq \gamma \leq 1$ est le coéfficient d'amortissement qui définit l'importance relative des récompenses présentes et futures. $E_{\pi}[.]$ désigne la valeur attendue étant donné que l'agent suit une politique π. La politique π est une suite qui indique pour chaque état quelle est l'action à prendre.

Ce problème est résolu en utilisant l'algorithme d'itérations sur les valeurs. Cet algorithme consiste à calculer la fonction valeur optimale V^* par évaluation successive de la fonction V_t selon le schéma d'itération $V_{t+1} = \mathcal{T}V_t$, où $\mathcal{T}$ est l'opérateur de Bellman [SB98]. Dans la partie expérimentale, on compare la politique à court terme ($\gamma = 0$) et la politique à long terme ($\gamma = 0.9$) en terme de qualité vidéo décodée.

4.2.3 Etude expérimentale

La performance du processus de filtrage de couches proposé est évaluée en utilisant la séquence Foreman au format QCIF à une cadence $F = 30$ images/seconde. Les tests sont effectués en utilisant le codeur H.264/SVC (JSVM 9.11) [VWS07]. Parmi les trois types de scalabilité (temporelle, spatiale, et en qualité), dans cette partie, nous avons considéré la scalabilité en qualité où le nombre de couches décodées impacte directement la qualité vidéo décodée.

L'évolution du PSNR pour la luminance du flux vidéo décodé pour les deux politiques à court et à long terme sont représentées sur la figure 7. En moyenne sur la séquence considérée, un gain d'environ 0.9 dB est obtenu avec la politique à long terme comparée à celle à court terme. Ce gain est principalement dû à une meilleure gestion de filtrage des couches scalables. De plus, les tampons sont mieux contrôlés avec la politique à long terme et cela avec moins de situation de débordement dans les tampons de post-encodage et moins de situations de vide dans les tampons du récepteur. L'impact des actions sur le niveau des tampons du récepteur est également analysé. Les résultats montrent une augmentation de la qualité vidéo lorsque l'information sur les états des tampons du récepteur est exploitée par rapport au cas où les états de ces tampons ne sont pas considérés dans le système de contrôle.

Afin d'évaluer la dépendance du processus de filtrage de couches proposé par rapport au débit de canal, des simulations complémentaires ont été réalisées avec différentes valeurs de débit instantané. Nous avons constaté que la politique à long terme améliore le PSNR moyen, même pour de faibles valeurs de débit de canal par rapport au débit moyen de codage de la séquence. Les ressources disponibles sont

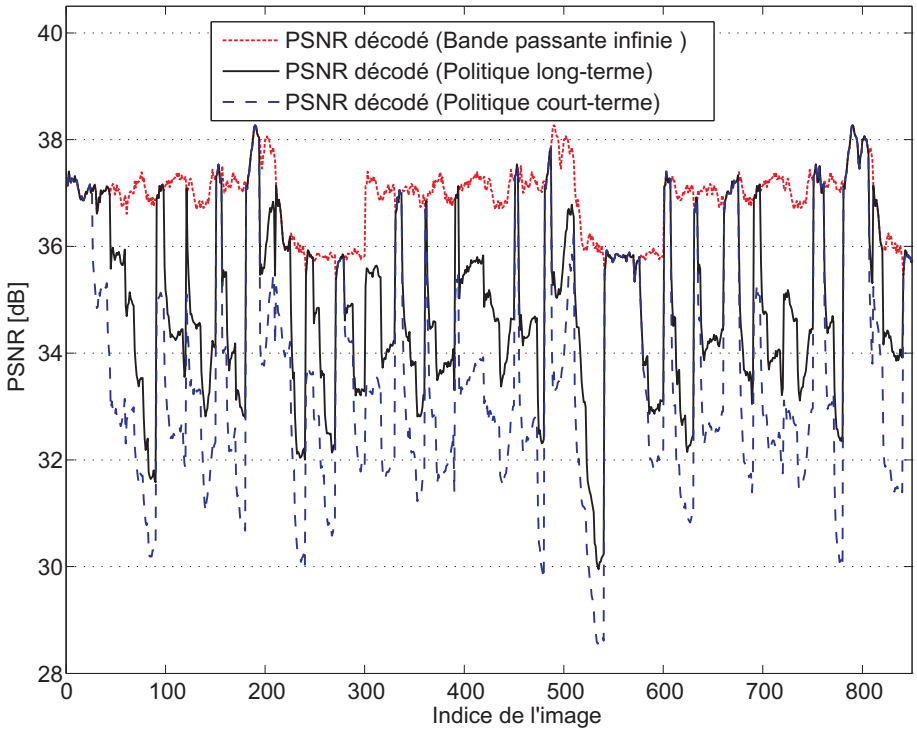


Figure 7: PSNR des séquences décodées avec les politiques à court et à long terme.

donc mieux exploitées avec l'approche à long terme. A débit de canal élevé, les deux politiques donnent des résultats similaires.

4.2.4 Conclusion

Nous avons considéré le problème de la transmission d'une vidéo scalable sur un canal sans fil afin de pouvoir contrôler l'envoi de la vidéo dans le réseau et de maximiser la qualité vidéo à la réception. Ce problème est formulé dans le cadre du PDM. Les résultats expérimentaux montrent que, grâce à une meilleure gestion des ressources et un meilleur contrôle des tampons, l'approche à long terme du PDM fournit une amélioration du PSNR moyen par rapport à l'approche à court terme. Le processus de filtrage proposé est effectué en utilisant un contrôle hors ligne. Toutefois, les caractéristiques des vidéos codées ainsi que l'état du canal sont variables dans le temps. Ces informations doivent être apprises en ligne pour tenir compte des variations dynamique du système. De plus, dans cette étude, l'état du canal ainsi que celui des tampons du récepteur sont considérés immédiatement disponibles chez le processus de filtrage de couche. Or, cette information envoyée par le récepteur arrive en général avec un retard. Ce dernier peut entrainer des problèmes d'instabilité dans le système de contrôle.

4.3 Apprentissage en ligne pour la transmission vidéo sur un réseau sans fil avec retour d'information retardé

Dans le chapitre 7, on s'intéresse au problème des retours d'information avec retard. Cette information concerne essentiellement l'état du canal envoyé par le récepteur vers le système de contrôle localisé dans le réseau. Pour cela nous proposons un

schéma de filtrage amélioré où un contrôle conjoint entre la couche application et la couche de contrôle d'accès au support (MAC) est considéré. Ce contrôle utilise implicitement des retours d'information de la couche MAC mais pas ceux de la couche application évitant ainsi l'utilisation des mesures retardées. Le problème de retard est traité en utilisant différents niveaux de connaissance de l'état du canal à savoir, l'état est immédiatement connu, l'état est connu avec un retard, et l'état est inconnu.

Là aussi le système de filtrage de couches est formulé et résolu dans le carde du PDM qui considère explicitement la coopération entre la couche application et la couche MAC, l'hétérogénéité des contenus vidéo, et les paramètres variables du canal. L'algorithme de filtrage des couches scalables a pour objectif de maximiser la qualité vidéo.

4.3.1 Description du problème

Le système de transport unicast considéré est représenté sur la figure 8. Le coeur du réseau se compose d'un serveur de streaming, un *proxy*, et une station de base. Les paquets sont transmis à travers un canal sans fil et reçus par un client mobile. Parmi les composantes de la station de base, nous considérons principalement le tampon MAC. La variation continuelle du contenu vidéo ainsi que des conditions du canal constituent une difficulté dans l'optimisation de la transmission vidéo. Une des méthodes utilisées pour aborder ce problème est d'apprendre en ligne la politique optimale à adopter dans les différentes situations du système avec des interactions continues entre le système de contrôle et l'environnement qui l'entoure.

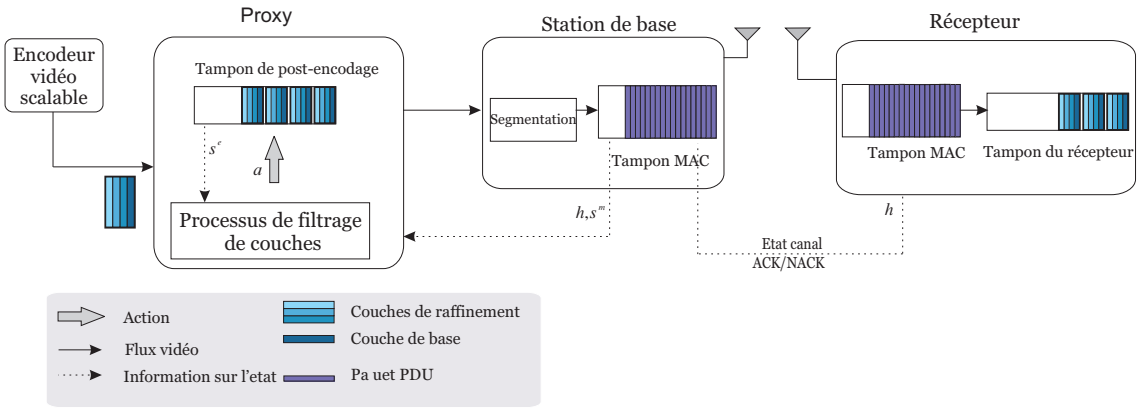


Figure 8: Système de transmission de vidéo scalable sur un canal sans fil.

4.3.2 Résolution du problème

Le processus de filtrage des couches doit maximiser la qualité vidéo du côté du récepteur. Pour résoudre ce problème on souhaite définir la fonction récompense uniquement à partir du PSNR résultant de l'action choisie. Cependant, il est difficile d'estimer de manière l'efficace le PSNR décodé au niveau du contrôleur en raison du retard causé par les tampons de la couche MAC et du récepteur.

Pour résoudre le problème du retard de transmission, le processus de filtrage de couche doit être en mesure d'estimer de manière efficace le PSNR décodé en se basant sur l'action choisie et l'état des tampons. Ainsi, la fonction de récompense à maximiser par le PDM, correspondant au PSNR estimé, est

$$r_t\left(\mathbf{s}_t, \mathbf{a}_t\right) = \left(1 - P_{\mathrm{d}}(\mathbf{s}_t, a_t)\right) \cdot r_t^q\left(s_t^I, \mathbf{a}_t\right) + P_{\mathrm{d}}(\mathbf{s}_t, a_t) \cdot (r_{t-1} - \lambda(s_t^I)), \tag{9}$$

tel que le terme $r_t^q\left(s_t^I, \mathbf{a}_t\right)$ correspond au PSNR estimé à partir des états du système et de l'action, $P_{\mathrm{d}}(\mathbf{s}_t, a_t)$ est la probabilité que l'image correspondant au PSNR estimé ne soit pas perdue, et $\lambda(s_t^I)$ est la baisse en PSNR dûe à la perte d'une image par rapport au PSNR de l'image précédente r_{t-1}. s_t^I correspond à l'état de l'image filtrée indiquant le type d'image (I ou P) incluant ainsi le niveau de priorité de l'image dans le système de filtrage.

Face aux caractéristiques variables dans le temps des données multimédia et des conditions de canal, nous considérons un algorithme d'apprentissage en ligne qui permet la mise à jour des politiques de décision en considérant l'impact de la décision actuelle sur les récompenses futures.

Un modèle de Markov du canal est considéré et trois hypothèses concernant la connaissance de l'état du canal sont considérées, à savoir, l'état est immédiatement connu, l'état est connu avec retard, et l'état est inconnu. Sans observation de l'état du canal, la commande exploite l'observation du niveau du tampon de la couche MAC uniquement.

4.3.3 Etude expérimentale

La performance du processus de filtrage de couche proposé a été évaluée sur les séquences Foreman et Mother & Daughter au format QCIF à $F = 30$ images/seconde. Les tests sont effectués en utilisant le codeur H.264/SVC.

La période temporelle à laquelle le système de contrôle opère est $T = 1/F$. Nous avons évalué la performance du processus de filtrage de couches proposé en utilisant les trois hypothèses concernant la connaissance de l'état du canal. Notre système est comparé à un extracteur de flux en bits basique indépendant du contenu (ou BBSE) utilisant également les trois hypothèses sur l'état du canal. Cet extracteur consiste à extraire des couches SVC selon un ordre de priorité précis. La hiérarchisation se fait en fonction des identificateurs de dépendance syntaxique des éléments: le dépendance_id, le temporel_id et le qualité_id. Le nombre de couches SVC à transmettre pour chaque image est sélectionné en fonction de l'ordre des priorités définies indépendamment du contenu de l'image.

Les figures 9 et 10 montrent la performance des trois hypothèses considérées concernant la connaissance de l'état du canal dans le processus de filtrage de couche proposé pour les séquences Foreman et Mother & Daughter utilisant la politique à long terme et comparé au processus BBSE pour différents débit canal.

On constate que dans le cas de connaissance retardé de l'état du canal, le système de filtrage proposé avec la politique à long terme surpasse le schéma BBSE en terme de PSNR moyen de 0.32 dB pour la séquence Foreman et 0.27 dB avec la séquence Mother & Daughter.

Lorsque l'état du canal n'est pas pris en compte dans le modèle du système, le schéma proposé avec la politique à long terme surpasse le schéma BBSE d'une

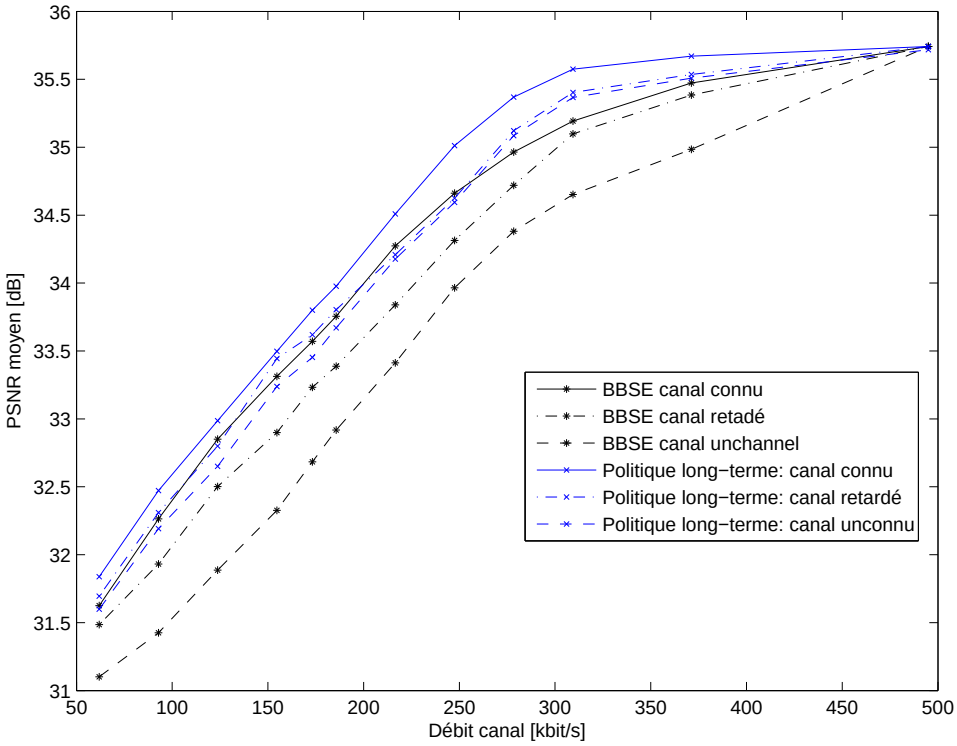


Figure 9: PSNR de la séquence Foreman lorsqu'une politique à long terme est appliqué et en considérant les trois hypothèses sur l'état du canal.

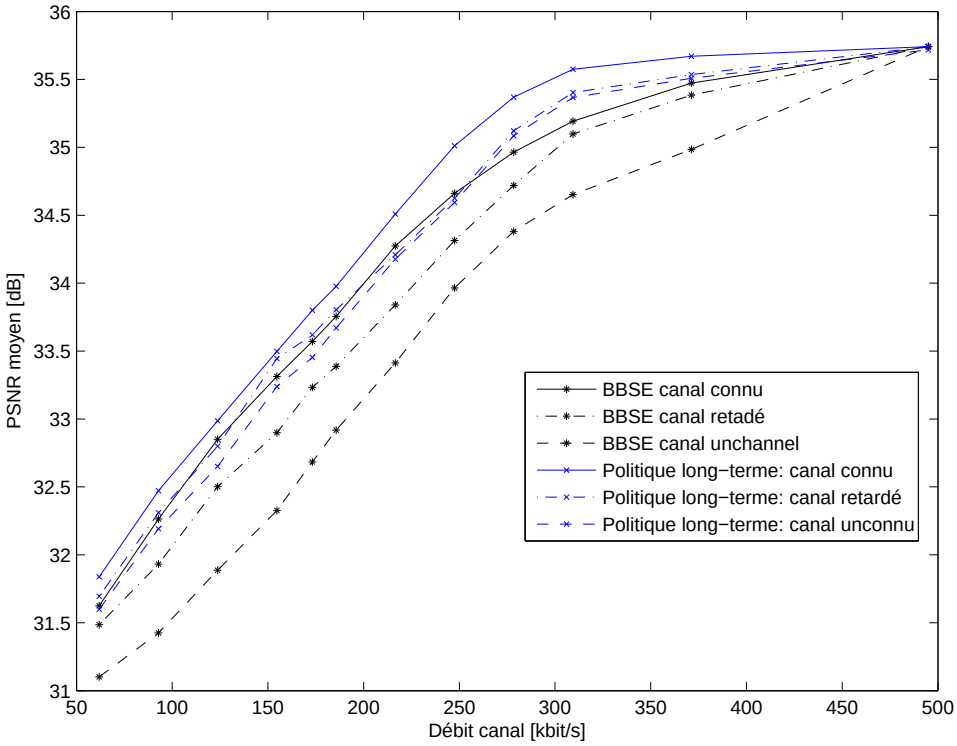


Figure 10: PSNR de la séquence Mother & Daughter lorsqu'une politique à long terme est appliqué et en considérant les trois hypothèses sur l'état du canal.

moyenne de 0.64 dB en PSNR pour la séquence Foreman et 0.56 dB avec la séquence Mother & Daughter. Quand l'état du canal est considéré immédiatement connu, les performances du schéma proposé et ceux avec schéma BBSE sont comparables.

Le gain en PSNR moyen obtenu avec le schéma proposé par rapport au schéma BBSE est principalement dû au fait que le schéma de filtrage exploite l'estimation de la contribution de chaque couche sur la qualité de la vidéo, contrairement au régime BBSE qui utilise uniquement les éléments de syntaxe de l'unité NAL sans prise en compte de l'impact que cela peut avoir sur la qualité résultante.

Dans nos simulations, nous avons considéré un modèle de Markov à 2 états pour modéliser les variations des paramètres du canal. Cette hypothèse peut être à l'origine de cette limitation de performance du système de filtrage proposé. Un gain supérieur peut être atteint en considérant un modèle de canal avec une granularité moins grossière.

4.3.4 Conclusion

Nous proposons une solution de filtrage de couches pour la transmission de la vidéo scalable sur un canal sans fil. Pour résoudre le problème de variations des conditions du canal, nous considérons un algorithme d'apprentissage qui permet une mise à jour en ligne de la décision de filtrage en fonction des nouvelles caractéristiques. Par ailleurs, face au problème du retard des retours d'information de l'état du canal, nous proposons un filtrage des couches scalable qui permet une gestion conjointe au niveau des la couche application et de la couche MAC, où des retours d'information de la couche MAC sont implicitement utilisées, mais sans utiliser ceux de la couche application.

Différents niveaux de connaissance de l'état du canal sont considérés. Les résultats expérimentaux montrent qu'avec des informations d'état du canal retardé ou sans l'état du canal, les performances du système de contrôle avec la politique à long terme n'est que légèrement dégradée par rapport au cas où l'état du canal est disponible, contrairement au schémas de base proposé par le JSVM où la dégradation du PSNR est plus importante. Cela permet de conclure que l'observation du niveau du tampon MAC fournit une estimation satisfaisante de l'état du canal.

5 Conclusions

Dans cette thèse, nous avons développé des systèmes de contrôle pour le codage et la transmission d'un ou d'un ensemble de flux vidéo sur un canal variant dans le temps. Les contraintes à soulever prises en compte dans nos systèmes de contrôle sont principalement axées sur l'utilisation efficace de la bande passante disponible, les contraintes de qualité vidéo, et le délai de transmission.

Tout d'abord, nous avons développé un système de MS qui permet à plusieurs flux vidéo d'être transmis en parallèle sur un canal de diffusion, tout en assurant une utilisation efficace de la bande passante disponible, une variation lisse de la qualité vidéo, une équité en qualité vidéo entre les programmes transmis, et un délai de transmission limité. Nous avons montré que grâce à une commande prédictive, les contraintes de qualité vidéo sont bien satisfaites. En outre, le délai de transmission est réglé à l'aide d'un contrôleur PID. Grace à la commande en boucle fermée, un

délai similaire pour tous les programmes et une utilisation efficace du débit de canal sont assurés. Nous avons ensuite proposé un schéma de MS distribué qui permet de contrôler les paramètres de codage de chaque codeur sans échanger des informations entre les serveurs. Ce contrôle se fait en deux étapes: un premier contrôle est effectué au niveau de chaque serveur vidéo indépendamment des autres afin d'ajuster son débit de codage; un deuxième contrôle, centralisé, est réalisé dans le réseau afin d'ajuster l'allocation de bande passante entre les programmes et assurer une bonne utilisation de la bande passante disponible tout en tenant compte de la contrainte d'équité en qualité.

Ensuite, nous avons développé une solution pour la transmission de vidéo scalable sur un réseau sans fil. La solution proposée repose sur un algorithme de filtrage de couches conçu à l'aide des outils d'optimisation stochastique. La variation des caractéristiques du canal sans fil représente une des difficultés représentée dans la résolution du problème.

Là encore, les contraintes considérées dans le système d'optimisation sont principalement axées sur l'utilisation efficace des ressources disponibles, une qualité vidéo maximale à la réception et un contrôle efficace des tampons. La solution proposée est représentée dans le contexte du PDM qui considère explicitement la coopération entre la couche application et la couche MAC, l'hétérogénéité des données vidéo, et la dynamique variable du débit canal. Le retard des retours d'information entre le récepteur et le système de contrôle utilisé pour suivre l'évolution l'état du canal est pris en compte. Différents niveaux de connaissance de l'état du canal sont considérés, à savoir, l'état est immédiatement connu, l'état est connu avec un retard, et l'état est inconnu. Même dans les deux derniers cas, la conception d'une loi de commande peut être résolue dans un cadre de PDM classique. Chaque niveau de connaisance est ensuite testé dans le processus de filtrage de couches proposé. Cela a permis de montrer que sans l'état du canal, le contrôle peut estimer efficacement ce dernier en observant seulement le niveau de la mémoire tampon du MAC.

Chapter 1

Introduction

1 Motivation

The increase of available bandwidth in wireless networks allows a larger diversity of services provided to users. Thus, Mobile television, video on demand (payed or via YouTube), telephony via data channel are applications that are currently emerging [Cis09]. Even if wireless resources are also expected to grow significantly with the availability of 4G networks, resources required to provide compressed streams compatible with large-display mobile receivers, such as smart-phones, or mobile tablets are growing even faster. Thus, with the increasing demand for always better quality programs, a proper management of the available bandwidth is required. Apart from the optimal use of the bandwidth, efficiently designing and implementing such delay sensitive multimedia applications on quality-constrained, heterogeneous systems is challenging due to their real-time constraints as well as the time-varying environmental dynamics experienced by the system (e.g. video source characteristics, user requirements, number of running applications, memory, channel conditions, etc.). The eventual success of video applications depends on the efficient management of system resources of both the transmitted data and the network such as bandwidth, delay, quality etc. To address these problems, novel network resource management techniques are required.

2 Challenges

In broadcast and multicast systems, video programs are transmitted over a shared channel. Apart from the optimal use of the available channel rate, statistical multiplexing (SM) systems may target the satisfaction of several constraints linked to the requirements of the delivered programs, including video quality and transmission delay, *e.g.* minimum quality, fairness, and smoothness, delay constraints. Finding an SM system able to satisfy simultaneously all these constraints in the context of video broadcasting is still a challenging task. This is mainly due to the non-stationary content of each program. Variations may be due, *e.g.*, to scene change or to high activity within a program. Moreover, most of the existing SM are performed in a centralized way where video servers share continuously information in order to satisfy quality fairness constraint. However, since video servers can be located in separate places, a decentralized rate control scheme is desirable.

In addition, most of existing video control systems and network resource management solutions rely on short term optimizations, which care only about the immediate performance of the system. However, in almost all resource and system management problems, the decisions made at the current time impact the immediate and future system performance. Acting on accurate predictions about the future dynamics can improve resource utilization, delivery delay, and video quality. In this dissertation, we interchangeably refer to this type of optimization as a predictive, long-term, or foresighted optimization.

In summary, finding new patterns for how innovating multimedia system and network resource management is necessary, where devices can exploit future information and autonomously learn on-line the unknown dynamics that they experience, and dynamically manage multimedia and network resources to obtain long-term optimal performance.

3 Contributions

In chapter 2, we define the several technical materials used throughout this dissertation including an overview of the different basic functional elements of the H.264/AVC and H.264/SVC encoders as well as rate-distortion models fundamental in the rate control process. In chapter 3, we focus on the inter-frame dependency specified in terms of impact of encoding parameters of a given frame over the next one. The dependency is interpreted theoretically by considering the quantization of an independent regime-switching autoregressive source model (IRS-AR) with variable quantization parameters. The following sections summarize the remainder of this dissertation.

3.1 Chapters 3&4: Statistical multiplexing over broadcast network

Statistical multiplexing of video contents aims at transmitting several variable bit rate (VBR) encoded video streams over a band-limited channel. Rate-distortion (RD) models for the encoded streams are often used to control the video encoders. Buffering at the output of encoders is one of the several techniques used to smooth out the fluctuating bit rate of compressed video due to variations in the activity of video contents. In chapter 4, a centralized statistical multiplexer is proposed where a closed-loop control of both video encoders and buffers is performed jointly. First, a predictive joint video encoder controller accounting for minimum quality, fairness, and smoothness constraints is considered. Second, all buffers are controlled simultaneously to regulate the buffering delays. This delay is adjusted according to a reference delay constraint. The main idea is to update the encoding rate for each video unit according to the average level of the buffers, to maximize the quality of each program and effectively use the available channel rate. Simulation results show that the proposed scheme yields a smooth and fair video quality among programs thanks to the predictive control. A similar buffering delay for multiplexed programs and an efficient use of the available channel rate are obtained thanks to the buffer management and to the predictive closed-loop control. In chapter 5, we propose a control system that allows performing statistical multiplexing of multiple

video stream in a decentralized way. We consider a media aware network element (MANE) fed by several remote video servers. The role of the MANE is to buffer the encoded video contents and to build a multiplex containing all video programs to be broadcasted over the broadcast channel. Each video server is controlled independently from the others, requiring no exchange between servers. The bandwidth allocation among programs is centralized and done within the MANE, but takes into account the quality fairness constraint. The MANE feds back to each video server the level of its associated buffer to help the remote video servers adapting their rate-distortion trade-off so that the buffer reaches some reference level.

3.2 Chapters 6&7: Scalable video transmission over Unicast network

While Chapters 4 and 5 focus on multiple video transport optimization over broadcast channel, Chapters 7 and 8 focus on resource management for scalable video transmission in wireless networks. After introducing the several notations and algorithms required to cast optimization problem into a Markov Decision Process (MDP) framework in Chapter 6, we address in Chapter 7 the problem of video on demand delivery over a time-varying wireless channel. Video quality optimization and buffer management are jointly considered for scalable video transmission to adapt to the changing channel conditions. A filtering algorithm among scalable layers is considered to maximize the decoded video quality at the receiver side while keeping a minimum playback margin. This problem is cast in the context of MDP which allows the design of foresighted policies maximizing some long-term reward. In Chapter 8, layer filtering process is performed while considering both Application and Medium Access Control (MAC) layers. As in Chapter 7, we formulate the video transmission problem as an MDP problem that explicitly considers the cooperation at the application layer and the MAC sublayer, the heterogeneous of the video data and the dynamically varying network conditions across time. To tackle the problem of feedback delay about the channel state information, various levels of knowledge of the state of the channel are considered, namely, immediately known state, known state with delay, and unknown state. When the channel state is delayed or unknown, the control has to rely on the observation of the level of the MAC buffer only. A challenging task in delivering multimedia data over wireless network is the dynamic characteristics of both the wireless channels and the multimedia source data. To overcome this problem, We consider learning technique which allows an on-line update of the filtering decision according to the new arriving characteristics.

Conclusions are provided in Chapter 9 as well as a discussion about future research directions.

Part I

Coding Tools for Video Rate Control

Chapter 2

Rate and Distortion Models for Video Rate Control

1 Introduction

In this chapter, we define several technical tools used throughout this dissertation. We propose an overview of the different basic functional elements of the H.264/AVC encoder as well as the fundamental features provided by the scalable encoder H.264/SVC extension of the H.264/AVC. We also discuss the different methods to perform rate control allowing the adjustment of the rate and the video quality according to the video content. The Rate-Distortion (R-D) trade-off control is usually based on R-D models. Thus, several existing R-D models are recalled as well as the inter-frame dependent models.

2 Video encoders

Video coding standards experienced significant development and improvement in video compression techniques. Following H:261 (1990), MPEG-1 Video (1993), MPEG-2 Video (1994), H:263 (1995- 1997), MPEG-4 part 2 (1998), the H.264 (2003) (also known as MPEG-4 Part 10/AVC - Advanced Video Coding (AVC)) is currently the newest video compression standard.

H.264 is the result of a joint project between the Video Coding Experts Group (VCEG) of the International Telecommunications Union (ITU-T) and the Moving Picture Experts Group (MPEG) of the ISO/IEC. H.264 standard was presented as a new element of the series of MPEG-4 standards. H.264 provides a high compression ratio for the video source information and further improve the efficiency of the communication systems. In fact, H.264 outperforms the previous video coding standards by utilizing a variety of temporal and spatial predictions. H.264 encoder can reduce the size of a digital video file of more than 50% compared to the MPEG-4 Part 2 [WSBL03], without the image quality being compromised. The resulting compressed file requires much less storage space and bandwidth when transmitted over the network. This also means that the video quality is much higher for a given bit rate. The rate and quality trade-off is one of the most important issue addressed in the rate control algorithms to adjust jointly the rate and the quality among video frames.

2.1 H.264/AVC

H.264 standards are deployed in various multimedia applications such as video conferencing, video storage, video-on-demand (VOD), digital television broadcasting, and Internet video streaming. Service providers and telecommunications companies are beginning to adopt H.264. H.264 standardization allows integrating new video compression standard into new electronic devices such as mobile phones and digital video players.

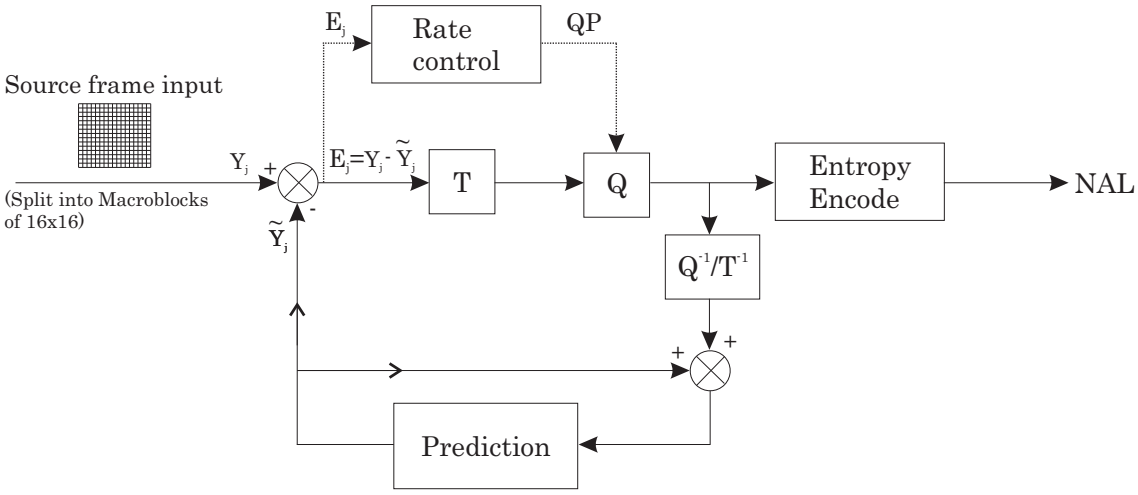


Figure 2.1: H.264/AVC Coding Scheme.

2.1.1 Encoder

The basic functional elements (prediction, transform, quantization, entropy encoding) are little different from previous standards (MPEG1, MPEG2, MPEG4, H.261, H.263); the important changes in H.264 occur in the details of each functional element. Figure 2.1 illustrates the functional elements and step in video compression. The hierarchy of a video sequence, from sequence to samples is given by: sequence → frame → slices → macroblocks → macroblock partitions → sub-macroblock partitions → blocks → samples.

Video frame is compressed using different algorithms centered mainly around the amount of data compression. The three major frame types used in the different video algorithms are I, P and B. They are different in the following characteristics:

- I-frames are the least compressible but do not require other video frames to decode, also called Intra coded frame.
- P-frames can use data from previous frames to decompress and are more compressible than I-frames also called Inter coded frame.
- B-frames can use both previous and forward frames for data reference to get the highest amount of data compression.

Frames can be grouped into Group of pictures (GoP). The GoP structure, specifies the order in which frame types are arranged. Each coded video stream consists of successive GoPs.

2.1.1.a Prediction

The encoder processes a frame of video in units of a Macroblock (MB) (16x16 displayed pixels). The purpose of the prediction process is to reduce spatial (intra mode) or temporal (inter mode) redundancies by exploiting correlation between adjacent MBs in a given frame or between MBs in successive frames in a GoP. Residuals are results of the subtraction of the predicted MB from the current MB. The H.264 encoder supports the prediction process with more flexible and more various prediction modes than those in previous standards. This flexibility and variability enables a more efficient video compression.

Intra prediction uses 9 prediction modes calculated for both 16x16 and 4x4 block sizes to predict the MB from surrounding, previously-coded pixels within the same frame.

Inter prediction supports motion compensation block sizes ranging from 16x16 to 4x4 luminance samples with many options between the two to predict pixels in the current frame from similar regions in previously-coded frames. The coefficients of the previously coded frame can be used as a prediction of current frame by Motion Compensation (MC). MC exploits high temporal correlation between successive frames of an image sequence and it mainly consists of two parts: motion estimation (ME) and prediction error coding. The displacement of objects between successive frames is first estimated (ME). The resulting motion information is then exploited in the inter-frame prediction. The prediction error is the difference between the current frame and the motion-predicted frame. Both the motion information and prediction error should be encoded and transmitted to the decoder.

2.1.1.b Transform (T) and Quantization (Q)

After spatial prediction, a block of residual samples is transformed using a 4x4 or 8x8 integer transform: an approximate form of the Discrete Cosine Transform (DCT). There are several features about the transform selected for the H.264 coding standard. Some of these features are:

- Transform is based on an integer transform (all operations can be carried out with integer arithmetic, without loss of accuracy).
- The inverse transform is fully specified in the H.264 standard [WSBL03] and if this specification is followed correctly, no mismatch between encoders and decoders occurs.
- The core part of the transform only requires additions and shifts.
- The entire process of transform and quantization can be carried out using 16-bit integer arithmetic and only a single multiply per coefficient, without any loss of accuracy.

The output of the transform, a block of transform coefficients, is quantized, *i.e.*, each coefficient is divided by an integer value. Quantization reduces the precision of the transform coefficients according to a quantization parameter (QP), where QP can take a value between 0 and 51 inclusive. The quantization is performed over a finite continuous interval leading to a lossy compression.

Setting QP to a small value leads to small compression and thus to little data loss. As QP increases in value the quantization process starts removing information. However, the encoder is designed to remove only the most insignificant details first and often this lost information is imperceptible to the human eye. As QP increases further towards the maximum value 51, more and more information are discarded, and video quality has to be sacrificed. The value of QP can be set as an input parameter or can be generated by the rate control algorithm in charge of satisfying a target bit rate. More details about the rate control process are provided in Section 2.3.

2.1.1.c Encoding

The quantized coefficients of the transform are scanned in one of the two different ways (zig-zag or field scan) depending on the prediction mode used for intra prediction process (field mode: field neighbors are used for spatial prediction, or frame mode: frame neighbors are used for prediction). This ordering is designed to order the highest-variance coefficients first and to maximize the number of consecutive zero-valued coefficients appearing in the scan. The ordered coefficients are then entropy coded to reduce the data size. Two modes of entropy coding are used in this standard: variable length coding (VLC) and binary arithmetic coding (BAC). In H.264/AVC, both of these designs are used in a context adaptive (CA) way, leading to the terms CAVLC and CABAC. The video coding process produces a number of values that must be encoded to form the compressed bitstream. These values include:

- quantized transform coefficients,
- information to enable the decoder to re-create the prediction,
- information about the structure of the compressed data and the compression tools used during encoding,
- information about the complete video sequence.

These values and parameters are converted into binary codes. Each of the two entropy coding modes produces an efficient, compact binary representation of the information. The encoded bitstream can then be stored and/or transmitted. The entropy encoded coefficients, together with side information are required to decode the MB (such as the MB prediction mode, quantization step size, motion vector information describing how the MB was motion-compensated, etc) form the compressed bitstream which is then passed to a Network Abstraction Layer (NAL) for transmission or storage. The NAL is designed to fit a variety of delivery frameworks (e.g., broadcast, wireless, storage media).

2.1.2 Decoder process

The video decoder receives a compressed bitstream from the NAL. Each of the system elements is entropy decoded to extract the information described above (quantized transform coefficients, prediction information, etc). Thus, each quantized transform coefficient is rescaled (inverse quantization Q^{-1}) then an inverse

transform (T^{-1}) is performed to re-create the original block. By adding the header information decoded from the bitstream, the decoder creates a prediction MB, identical to the original prediction MB formed in the encoder.

2.2 H.264/SVC

2.2.1 Overview

Scalable Video Coding (SVC) [SMW07] was defined as an amendment of H.264/AVC, providing efficient scalable representation of video by flexible multi-dimensional resolution adaptation. It was proposed to provide network-friendly scalability at a bit stream level and a very simple stream truncation allowing a significant increased error robustness and a moderate increase in decoder complexity relative to single-layer H.264/AVC.

Scalability here refers to the ability of a compression algorithm of representing an encoded source over multiple hierarchical bitstreams. Among these, a base layer (BL) is independently decodable from others and allows the reconstruction of data at a minimum level of quality. This BL should be H.264-AVC compatible which requires that the coding techniques and the different compression level are the same as those used in the H.264/AVC standard. Adding enhancement layers (EL) allows increasing the quality of the decoded stream. The H.264/SVC coding scheme for two scalable layers is represented in Figure 2.2.

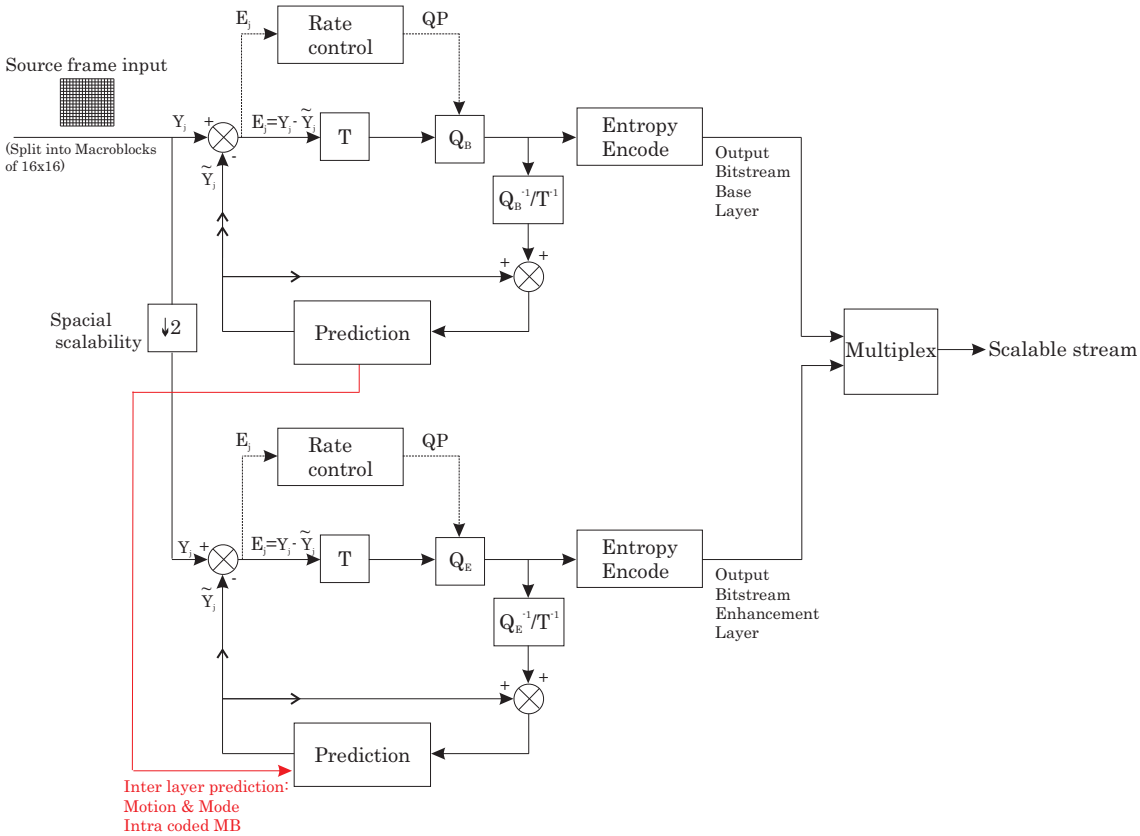


Figure 2.2: H.264/SVC Coding Scheme.

Unlike previous solutions, SVC provides a high degree of flexibility in terms of scalability dimensions supporting:

- Temporal scalability: the ability to present the video at different frame rates,
- Spatial scalability: the ability to present the video at different spatial resolutions,
- Signal-to-noise ratio (SNR) or Quality scalability: the ability to present the video at different quality levels (*i.e.*, bit rates).

2.2.1.a Temporal scalability

The temporal scalability consists in transmitting a bit stream over a set of access units (AU) partitioned into a temporal BL and one or more temporal ELs. Each temporal layers is identified by a temporal layer identifier T, which starts from 0 for the BL and is increased by 1 from one temporal layer to the next. Temporal scalability can generally be enabled by restricting motion-compensated prediction to reference frames with a temporal layer identifier that is less than or equal to the temporal layer identifier of the frame to be predicted.

2.2.1.b Spatial scalability

Spatial scalable coding consists in transmitting a bit stream over layers, each layer corresponds to a spatial resolution and is referred by a dependency identifier D. The dependency identifier for the BL is equal to 0, and it is increased by 1 from one spatial layer to the next. In each spatial layer, motion- compensated prediction and intra-prediction are employed as for single-layer coding. But in order to improve coding efficiency compared to simultaneously transmitting different spatial resolution, additional inter-layer prediction mechanisms are considered.

2.2.1.c SNR scalability

SNR or Quality scalability allows a frame to be coded with identical frame sizes for BL and EL but providing a graceful degradation of quality among SNR layers at the same spatial/temporal resolution. SNR scalability offers various granularity for different applications: Coarse-Grain quality Scalability (CGS), which supports bit rate adaptation at the level of SNR layers, and Medium-Grain quality Scalability (MGS), which supports bit rate adaptation at the frame level. In the first case, the SNR scalability provides only limited granularity, since switching between CGS layers is only supported at Instantaneous Decoder Refresh (IDR) frames. In order to increase the granularity of quality, SVC provides the possibility to use the quality identifier Q for quality refinements.

With the MGS scalability, any EL can be discarded from a quality scalable bit stream. Nevertheless, if the EL with the highest available quality is always applied as a reference for inter-frame motion prediction, discarding an MGS layer will introduce a drift effect between motion compensation prediction loops at the encoder and the decoder. Such a drift does not occur if motion compensation is only performed using BL as reference at the expense of decreasing the coding efficiency. To control

the drift and achieve high coding efficiency, MGS uses the key frame concept, as shown in the Figure 2.3 (d) where bolt arrow refers to inter frame prediction and dotted arrow refers to inter layer prediction. Frames that only use the BL frames for prediction are referred to as key frames, which requires both lowest quality and highest quality layers to be reconstructed; K denotes the key frames. Inter frame prediction is performed using the highest quality layer to predict non key frames. A flag is transmitted for each frame that identifies whether the BL or the EL of the reference frame is employed for motion-compensated prediction.

Various concepts for trading off EL coding efficiency and drift for packet-based quality scalable coding are possible. These concepts are summarized in Figure 2.3.

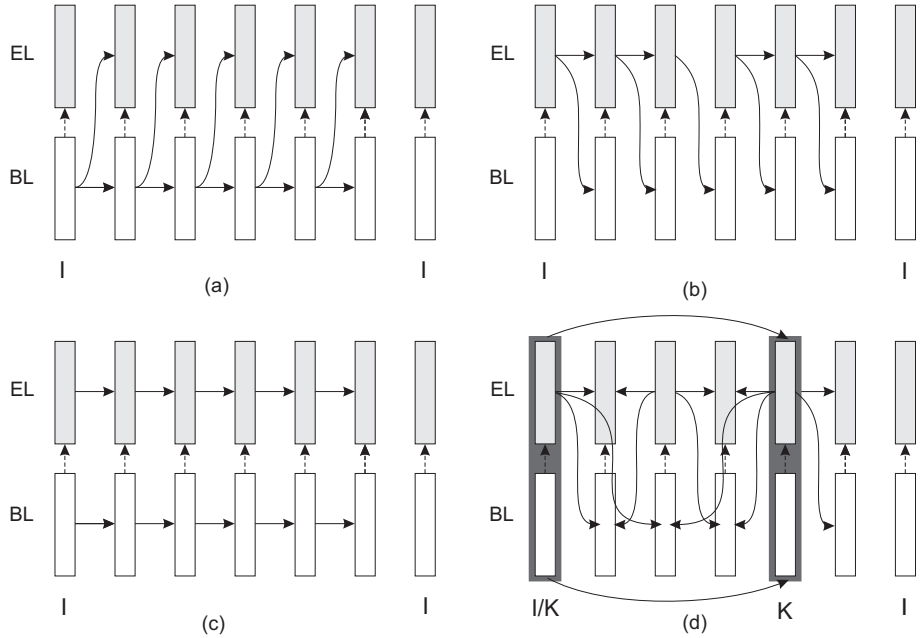


Figure 2.3: Prediction structure for SNR scalability: (a): SNR scalability in MPEG-4 Fine grain scalability (base layer only control); (b): SNR scalability in MPEG-2 (Enhancement layer only control); (c): SNR scalability in H.262/MPEG-2 (Two-loop control);(d): Key picture concept of H.264 scalable extension for hierarchical prediction structure.

In the SVC extension of H.264/AVC, the temporal, spatial, and quality scalability can be combined. Since the support of quality and spatial scalability usually comes along with a loss in coding efficiency relative to single-layer coding, the trade-off between coding efficiency and the provided degree of scalability can be adjusted according to the application.

2.2.2 Applications of the SVC encoded stream

The H.264/SVC provides various tools for reducing the loss in coding efficiency relative to single layer coding. The most important differences are as follows.

- The possibility to employ hierarchical prediction structures for providing temporal scalability with several layers.

- New methods for inter-layer prediction of motion and residual improving the coding efficiency of spatial and quality scalable coding.

- The concept of key pictures for efficiently controlling the drift for quality scalable coding with hierarchical prediction structures.

SVC replaces the all or nothing approach to video compression (shared by MPEG4 and conventional H.264) with a layered, scalable approach. In an SVC encoder, ELs are processed after the BL to produce a higher frame rate or higher resolution version of the video stream using the BL as reference. This technique of using previously encoded information to guide subsequent encoding reduces the overhead that would otherwise be incurred in a multi-encode system. An SVC video stream is 10-20% [CLS08,SRH10] larger than the size of the largest stream it carries when encoded in AVC with the same quality. Thus, when one SVC stream is sent instead of multiple individual video streams, a lot of bandwidth and storage space are saved.

The advantage of this approach is that a client device can decode the received stream, starting with the BL, and then decode incremental information from subsequent layers until the desired frame rate and resolution is achieved. A device having a lower resolution display or less compute power might decode after the first few layers. A higher powered or high definition client device might decode all of the layers as they arrive obtaining the video at full resolution and frame rate. This characteristic of SVC streams facilitates the adoption of high definition cameras whose streams would otherwise need to be re-encoded for legacy devices. Another advantage to this approach is that a multi-layered stream can simply be truncated to yield a decodable stream with lower resolution and frame rate. This can be done within the network itself, with the stream being truncated as it passes from a high bandwidth link to a lower bandwidth link. In this way, the stream is sized to match network bandwidth and yield video with reduced resolution or frame rate without having to decode the stream. This is a major improvement over the alternative, which requires a server in the network to decode the stream, scale the decoded video, and then re-encode the video as it is forwarded.

Combined scalability is highly desired for application such as surveillance, in which the video content will not only be viewed on different display systems, ranging from small screen video phones to high definition monitors, but also need to be stored and archived. The high resolution/quality part in the bit-stream could be deleted after sometime and only the low quality copies to be stored for archiving purpose.

2.3 Rate control

Among the various problems raised by the efficient delivery of video contents via a time varying channel, the control of the encoding rate is a challenging task. The major purpose of rate control is to regulate the bit stream according to the available bandwidth and to one or several buffers involved in the video encoding and transmission process while keeping the video quality as high as possible. This control has to cope (i) with a time-varying quality of the channel which results in variations in the channel capacity and (ii) with changing complexity of the scenes to be encoded. Quality constraints on the compressed video (limited distortion, smooth variations

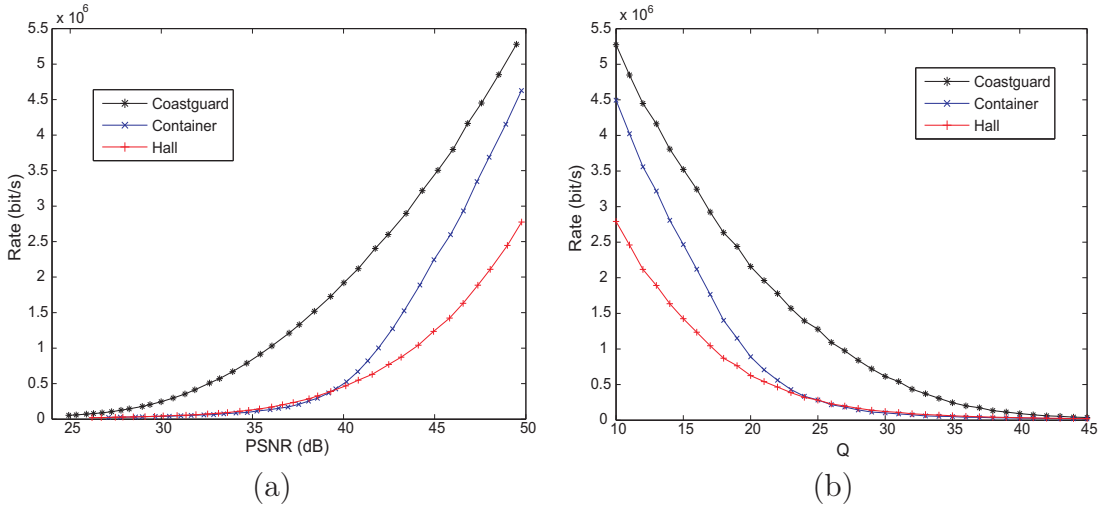
of quality) increase the difficulty of the rate control. Knowledge of R-D characteristic of the encoded streams is essential to perform a good rate control algorithm. Thus, the availability of well-tuned R-D models for each stream is very useful to satisfy the previously-mentioned constraints.

2.3.1 Generality

Video compression is based not only on the compression of the redundancy (temporal and spatial) but also on the compromise between coding efficiency and the video quality. Rate control is one of the video coding steps that ensures a trade-off between video quality and the coding rate. Although rate control does not belong to the normative part in video coding standards, it represent an essential part of the coding process in many applications to satisfy quality and/or buffers constraints.

The rate control in H.264/AVC encoder can be performed at four different granularities: GoP level, frame level, slice level, and MB level [DOC+06, YW07, LOHK08, PBM+10]. There are a number of encoding parameters (QP, size of blocks, motion compensation, etc.), that more or less influence the rate-distortion (R-D) values and that can be selected by the rate control for every video unit. Among these parameters, the QP has the most important impact on the R-D values. The task of rate controller is to determine a set of coding parameters such that a certain R-D trade-off is achieved for a given decoder.

In Figure 2.4(a) the encoding rates of the first GoP (15 frames) of three video sequences (Coastguard, Container, and Hall in QCIF format), characterized by different video activities, are represented as a function of the video quality measured using the Peak Signal to Noise Ratio (PSNR) (an objective quality metric). Figure 2.4(b) corresponds to the encoding rate as a function of the QP for the three sequences. Compromise between coding efficiency and the video quality is directly linked to the choice of the QP. As shown in Figure 2.4(b), increasing QP degrades the quality of the frame but decreases the coding rate.



(a) (b)

Figure 2.4: (a): Encoding rate as a function of the PSNR; (b): Encoding rate as a function of the QP for different video sequences (Coastguard, Container, and Hall).

The problem of optimum bit allocation for inter modes in a motion estimation and successive MB mode decision process is considered in [WSJ+03]. During the

encoding process, all coding modes of every MB are examined and the resulting rates (R) and the distortions (D) are calculated. The mode that has the minimum $\mathcal{J}$ is selected as the optimum mode for every MB:

$$\mathcal{J} = D + \lambda R. \tag{2.1}$$

In this method, the Lagrangian multiplier λ is first calculated with an empirical formula using the selected QP denoted by Q for every MB:

$$\lambda = 0.852^{\frac{Q-12}{3}}. \tag{2.2}$$

Rate control in the H.264/AVC encoder is conducted via controlling for instance the QP and adjusting the Lagrange parameters accordingly using (2.2). Rate control can be performed in a closed loop or in an open loop control. A closed loop rate control mechanism is used in applications where a constant bit rate is targeted. To achieve a more or less constant bit rate, by specifying the target bit rate, QP values of each MB are dynamically varied based on the estimate of the source complexity. This method is also referred to as Constant Bit Rate (CBR) coding scheme. Nevertheless, using CBR scheme the target rate can be achieved but without any consideration about the complexity and the quality of the successive frames in the video sequence. In the open loop control, the uncompressed video and the QP values are supplied by the user. As the source video is encoded, a fairly constant quality video sequence can be obtained, but the bit rate may vary. This is due to the source complexity that continually changes in a real video sequence. This method is referred to as the Variable Bit Rate (VBR) coding scheme.

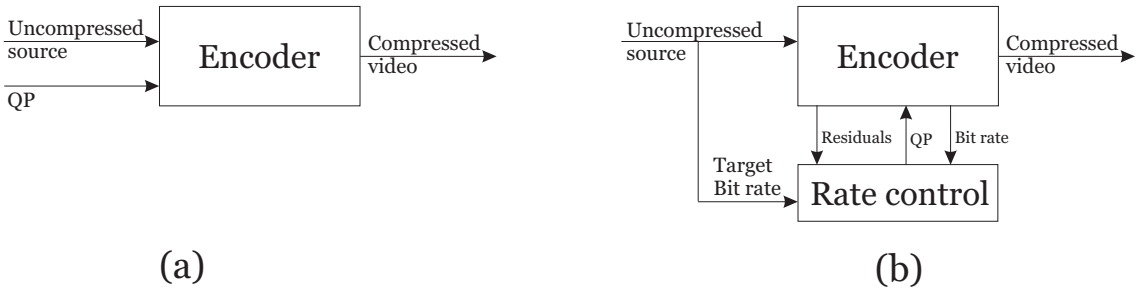


Figure 2.5: Rate control in an Open Loop Rate control (a), and in a Closed Loop Rate control(b).

Rate control can be considered also for H.264/SVC. The rate control can be applied on temporal [XMZG05], spatial [LCGK10] or SNR [LLS08] layers. In fact, scalable video can be conveniently delivered in heterogeneous networks with varying client display resolutions, transmission bandwidths, and network conditions. The H.264/SVC bit allocation algorithm can provide a good way to use the available resource and offer higher quality video to all clients of various spatial resolutions. The target bit rate can be achieved by varying the encoding parameters as in the rate control of non scalable stream or by performing some packet filtering process [SVHR10, MK10]. In this case, the number of transmitted layers at each frame is the control parameter.

3 Rate Distortion models

In this section, several state-of-the-art R-D models are recalled. Obtaining models for the R-D characteristics of the video sequence to be encoded is instrumental to build an efficient rate control. Among the wide variety of available models, one may identify the *independent* and *dependent* R-D models.

3.1 Independent models

In the first family of models, the R-D characteristics of each frame, or GoP, are assumed to be independent of those of the others frames, or GoPs. One gets then simple parametric models with few tuning parameters. Such models are quite efficient to represent the R-D characteristics of Intra-encoded frames, *i.e.*, frames encoded without reference to any others frames, or GoP. In this section, some previous independent R-D models are recalled. Two of these models are evaluated experimentally. Our goal is to get a simple R-D model for Intra coded unit with a minimum parameters to facilitate the rate control.

3.1.1 Some previous results

A brief review of existing independent R-D models is proposed in this section.

Most standardized video coders involve uniform fixed-step scalar quantization in the transformed domain. For Gaussian sources with zero mean and variance σ^2, the following R-D function is defined

$$D(R) = \sigma^2 2^{-2R}, \tag{2.3}$$

see [GN98]. Nevertheless, this simple model may only be used to estimate the R-D performance of the texture quantization process. Video coders involve other operations such as motion compensation or packetization of data, which contribution to the total rate is much more difficult to evaluate.

To define R-D models, various probability distributions of the DCT coefficients was considered, see [BLL00, TDP00, KAM05]. In [BBPR98], it has been demonstrated experimentally that full frame DCT coefficients can be modeled by a Laplacian density function, while in [Mul93] modeling DCT coefficient with a generalized Gaussian distribution is shown to be more accurate than with a Laplacian distribution. In [KAM05] a Cauchy density based R-D model for quantized DCT coefficients is proposed and analyzed in a frame bit allocation application for H.264/AVC video coding.

Based on these different analyses, several parametric models have been proposed to represent the R-D behavior of video coders.

A more sophisticated model than that in (2.3) is proposed in [MVHB04a] for Intra coded frame

$$D(R) = \sigma^2 \exp(-\beta R^\gamma), \tag{2.4}$$

The two parameters β and γ are introduced based on experimental observations to overcome the issue of model assumption violation due to coarse quantization using the Gaussian model.

A three-parameter R-D model is considered in [ZAPS+07]

$$D(R) = D_0 + \frac{\theta}{R - R_0}, \tag{2.5}$$

where the values of the parameters depend on the coding scheme and the content of the video.

Linear [HM02] and *quadratic* [LZ09] models are proposed to evaluate the rate as a function of ρ, the proportion of null coefficients of a quantized block in the transform domain. The quadratic R-D model is the following

$$\begin{array}{rcl} R(\rho) & = & \theta_1(1-\rho) + \theta_2(1-\rho)^2 \\ D(\rho) & = & \sigma^2 e^{-\alpha(1-\rho)}, \end{array} \tag{2.6}$$

where σ^2 is the frame variance and θ_1, θ_2, and α are tuning parameters. To be used, these models require the dependence of ρ with the value of the encoding parameters, as shown in [RBG08].

A piecewise Intra frame model is proposed in [Lin97]. Authors measure some points in $R(Q)$ and $D(Q)$ curves. The choice of these points is more frequent for small values of QP in order to follow the rapid variations of the curves. Piecewise cubic or linear interpolation is then used to estimate the rate and distortion for the remaining values of QP. The following R-D model is proposed

$$\begin{array}{rcl} R\,(Q) & = & a_{ij}Q^3 + b_{ij}Q^2 + c_{ij}Q + d_{ij} \\ D\,(Q) & = & e_{ij}Q + f_{ij}, \end{array} \tag{2.7}$$

where $\{a_{ij}, b_{ij}, c_{ij}, d_{ij}, e_{ij}, f_{ij}\}$, are model parameters for the $i-$th piecewise interval of $j-$th frame.

Inspired from [LGP+03], [WK07] proposed the following model for H.264/AVC at the MB level

$$\begin{array}{rcl} R(Q_{\text{step}}) & = & \left(\frac{a_1}{Q_{\text{step}}} + \frac{a_2}{Q_{\text{step}}^2} + a_3\right)(a_4 M + a_5) \\ D(Q_{\text{step}}) & = & a_6 Q_{\text{step}} \end{array}$$

where $\mathbf{a} = (a_1, \ldots, a_6)$ is the vector of parameters, Q_{step} indicates the quantization step obtained from the quantization parameter Q as $Q_{\text{step}} = 2^{(Q-4)/6}$ and S is the Mean Absolute Difference (MAD) of the collocated MB in the previous frame. To be adjusted, all these models need at least as many encoding trials (ET) at different values of Q as parameters to identify. Both models in (2.7) and in (2.8) require a large number of ET to be accurately tuned.

A zero-mean Cauchy distribution i used in [SSWL06] to fit the video signals. The entropy and distortion are calculated and then simplified based on experimental observations showing a linear relation between $\log(Q)$ and both $\log(R)$ and $\log(D)$. Thus, the following quadratic R-D model is proposed

$$\begin{array}{rcl} R\,(Q) & = & aQ^{-\alpha} \\ D\,(Q) & = & bQ^{\beta}. \end{array} \tag{2.8}$$

Using this model, only two ETs are needed to estimate the model parameters.

In [MGL05] and [YDXD05], based on some experimental observations, the following models for the PSNR (in dB), the distortion, and the rate as a function of Q have been proposed

$$\begin{aligned} P(Q) &= a_{\mathrm{P}} Q + b_{\mathrm{P}}, \\ D(Q) &= a_{\mathrm{D}} \exp(b_{\mathrm{D}} Q), \\ R(Q) &= a_{\mathrm{R}} \exp(b_{\mathrm{R}} Q). \end{aligned} \tag{2.9}$$

This model provides a good fit at a GoP level as illustrated in [YDXD05]. As in (2.8), here parameters $(a_{\mathrm{P}}, b_{\mathrm{P}}, a_{\mathrm{R}}, b_{\mathrm{R}})$ can be estimated using two ETs.

Since ETs introduce additional encoding delay, one targets using the minimum ETs when estimating the R-D model parameters. Thus, we are interested in models involving the least parameters. In the next section, we compare model performance of both the quadratic model in (2.8) and the exponential model in (2.9).

3.1.2 Models evaluation

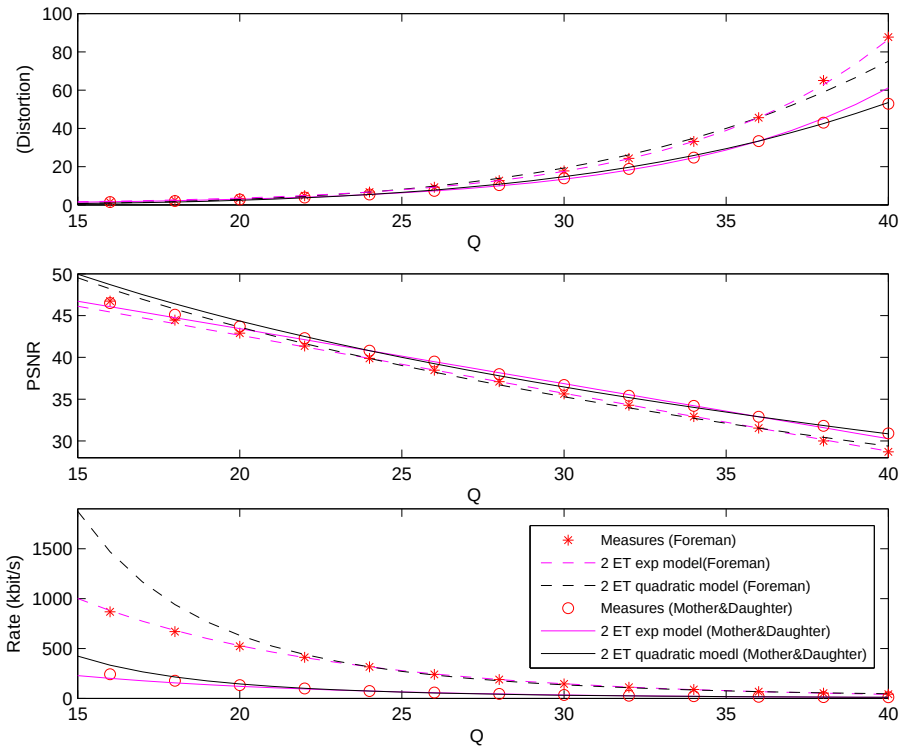


Figure 2.6: Performance of the quadratic model in (2.8) and the exponential Distortion, PSNR, and Rate models in (2.9) fitted with 2 ETs using Foreman and Mother&Daughter sequences.

We performed some tests using the quadratic models in (2.8) and the exponential models in (2.9) using some video sequences with different characteristics and contents. Figure 2.6 shows $D(Q)$, $P(Q)$, and $R(Q)$ functions for the first GoP of fifteen frames of *Foreman.cif* sequence and for the fifth GoP of seven frames of *Mother& Daughter.qcif* sequence.

For both quadratic and exponential model, parameters are estimated using two ETs with rwo values of QP $Q = (25, 35)$. From Figure 2.6, one suggests that the

exponential model accurately fits the actual evolution of the distortion (D), the PSNR (P), and the rate (R) independently of the value of the QP. The quadratic model fits less the actual evolution of the D, P, and R especially for small value of QP.

Table 2.1, illustrates the correlation coefficients r between the measured D, P, and R values and the estimated ones for both quadratic and exponential models using Foreman and Mother&Daughter sequences calculated as

$$r(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}}. \tag{2.10}$$

Where X is the vector of sample of the measured D, P, or R values (here 13 values) and Y is the vector of estimated D, P, or R values using the quadratic and the exponential models. Cov means covariance and Var means variance.

From Figure 2.6, on can see that the exponential R-D model can accommodate to a large quantization range, from low rate end to high rate end.

	Quadratic (2.8)			Exponential (2.9)		
	Distortion	PSNR	Rate	Distortion	PSNR	Rate
Foreman	0.831	0.995	0.906	0.999	0.998	0.999
Mother& Daughter	0.872	0.881	0.948	0.996	0.996	0.997

Table 2.1: Correlation coefficients for the Distortion, PSNR, and the Rate curves using quadratic and exponential model on Foreman and Mother&Daughter sequences.

3.2 Dependent models

It is well known that the quality at which a first frame is encoded impacts significantly that of the next frames when they are encoded with the first one as reference, as is done in most video coding standards based on the predictive coding principle [Say05], such as H.26X or MPEG X video coders.

Based on the independent piecewise model in [Lin97], the influence of the QP of reference frame denoted by Q_{j-1} and that of the predicted frame denoted by Q_j on the rate and the distortion are analyzed. The following linear distortion model is proposed

$$\begin{array}{lcll} D_j\left(Q_{j-1},Q_j\right) & = & \alpha - \beta\left[D_{j-1}(Q_j) - D_{j-1}(Q_{j-1})\right] & \text{if } Q_{j-1} \leq Q_j \\ D_j\left(Q_{j-1},Q_j\right) & = & \alpha & \text{else.} \end{array} \tag{2.11}$$

where α and β are parameters determined by encoding and measuring the distortion at two values of Q_{j-1}. $D_{j-1}(Q)$ is the distortion value obtained using the independent model in (2.7) and the quantization parameter Q.

Three rate models are proposed depending on the Q_{j-1} values,

$$\begin{array}{lcll} R_j\left(Q_{j-1},Q_j\right) & = & R_j(Q_1,Q_j) & \text{if } Q_{j-1} \leq Q_1 \\ R_j\left(Q_{j-1},Q_j\right) & = & \frac{R_j(Q_1,Q_j)(D_{j-1}(Q_2)-D_{j-1}(Q_{j-1}))}{D_{j-1}(Q_1)-D_{j-1}(Q_2)} & \\ & + & \frac{R_j(Q_2,Q_j)(D_{j-1}(Q_{j-1})-D_{j-1}(Q_1))}{D_{j-1}(Q_1)-D_{j-1}(Q_2)} & \text{if } Q_1 < Q_{j-1} \leq Q_j \\ R_j\left(Q_{j-1},Q_j\right) & = & R_j(Q_2,Q_j) & \text{if } Q_{j-1} > Q_2. \end{array} \tag{2.12}$$

where Q_1 and Q_2 are QPs corresponding to two measurement points considered from the experimental curves of the rate as a function of the previous QP. Nevertheless, the two parameters Q_1 and Q_2 should depend on the value of Q_j in order to accurately fit the rate behavior. A fixed values of Q_1 and Q_2 for all Q_j values may lead to important discrepancies between the actual rate and the model rate.

In [MVHB04a], different from the R-D model for Intra coded frames reported in (2.4), a R-D model is also introduced for Inter-coded frames

$$D_j(R_j) = \left(\sigma_j^2 + \alpha_j D_{j-1}(R_{j-1})\right) \exp(-\beta_j R_j^{\gamma}), \tag{2.13}$$

where σ_j^2 is the total variance of the j-th frame, and the term $\alpha_j D_{j-1}$ models the dependence between the current and the previous frame that captures the quantization error propagation effect caused by motion compensation. The propagation ratio α_j is determined by the amount of motion compensation.

To achieve an inter-frame dependency based budget allocation, the percentage of skipped MB in a frame is employed as the quantitative measure of the Inter-frame dependency in [LHK09], which forms the following dependent model

$$\begin{aligned} D_{j+1} &= SD_j + (1-S)\bar{S}_{j+1} \\ &= S\sigma_j^2 2^{-2uR} + (1-S)\bar{\sigma}_{j+1}^2 2^{-2(1-u)R}, \end{aligned} \tag{2.14}$$

where $\bar{S}_{j+1}$ is the average distortion for the non-skipped region in the $j+1-$th frame, S is the percentage of skipped macroblocks, σ_j^2 and $\bar{\sigma}_{j+1}^2$ are the standard deviations of motion compensated residues in the $j-$th frame and the non-skipped region of the $j+1-$th frame respectively, R is the total rate budget, and $u \in [0,1]$ represents the budget allocation scheme.

Models in (2.13) and (2.14) can be used for budget allocation rate control system, however, since the encoding parameters are not involved in these models, one can not use it in rate control system requiring the encoding parameters as actual control input. Moreover, the dependent R-D models proposed in [LO98, MVHB04a, LHK09] are based on experimental analysis. In [WSMG08] a theoretic analysis of optimal bit allocation in prediction-based video coders is presented. A Laplacian model is used to describe the distribution of DCT coefficients of residual frames and to analyze the dependency between reference frames and predicted frames at different bit rates. Based on these analysis, authors suggested that the difference of QPs between reference frame and predicted frame should not be too large. Furthermore, if there are more frames predicted by the reference frame, more bits should be allocated to it.

4 Conclusion

An overview of the different basic functional elements of the H.264/AVC encoder are presented in this chapter. The fundamental features of the scalable encoder H.264/SVC are also recalled. A special focus is devoted to the rate control process. This mechanism will be used later in this dissertation where we propose a statistical multiplexing system of video contents based on the R-D trade-off. Thus, we discussed the different methods allowing the adjustment of the video quality according to the bit rate. Several state-of-art independent and dependent R-D models are also

recalled. Experimental comparison between the independent exponential and the independent quadratic R-D models reveals that the exponential model is more accurate in the rate control system using large range of QP. Thanks to its simple form and to its accuracy, we propose to use the exponential model in the next chapters in the proposed R-D control systems.

Chapter 3

Independent Regime-Switching Auto-Regressive Model

1 Introduction

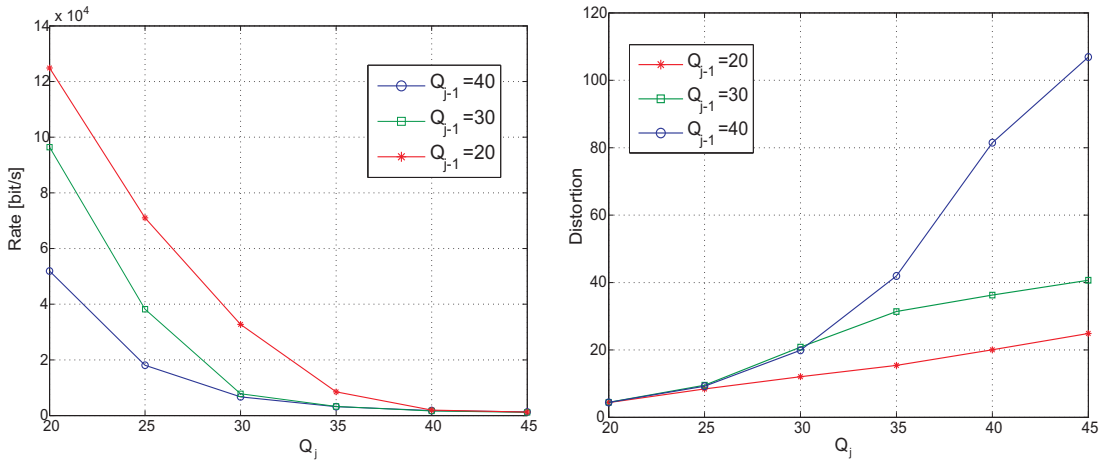
In this chapter, we focus on the construction of an efficient dependent Rate-Distortion (R-D) model for video coders involving motion compensation and texture encoding such as H.264/AVC. The dependency between a transform coefficient of the frame to be encoded and that of its prediction obtained from past (uncoded) frames is first examined using some experimental observations. The coefficients dependency is then described by an Independent Regime-Switching Auto-Regressive (IRS-AR) model [BP66]. This model allows representing coefficients linked to parts of the frame where motion compensation performs well (via the AR part), and coefficients related to parts not well predicted (with the regime switching part). The proposed dependent IRS-AR model has been published in ICASSP conference paper [CSK10a] and in the IEEE Vehicular technology Magazine [CSK09d].

2 Experimental observations

To determine whether the independent models may be extended to P frames or not, Figure 3.1 shows experimental R-D curves obtained for the second P frame of *Foreman* in QCIF format with H.264/AVC in baseline profile. Three QPs Q_{j-1} for the previous reference I frame have been considered and the QP Q_j for the P frame changes.

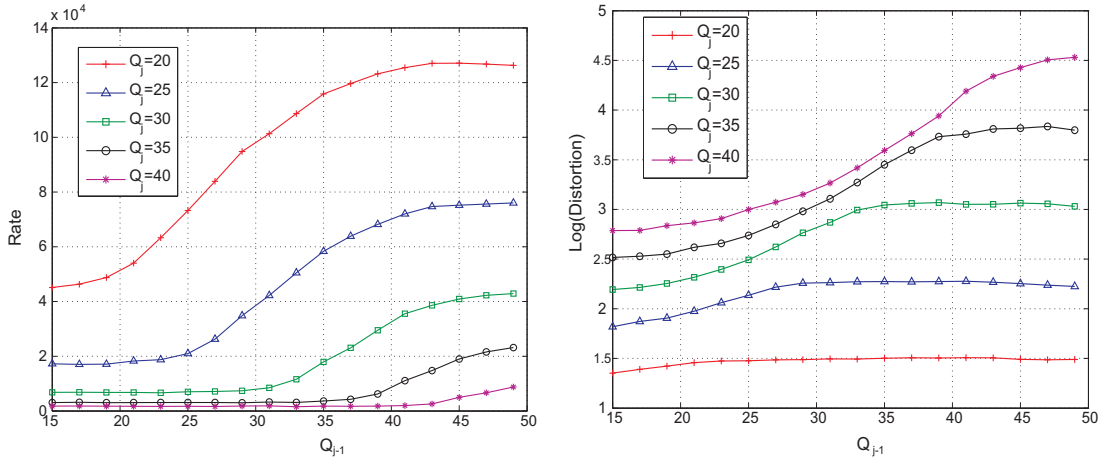
Figure 3.1(a) shows the rate of the j-th frame as a function of Q_j, parameterized with Q_{j-1}, for the first P-frame in Foreman sequence in QCIF format encoded with H.264/AVC in baseline profile. Figure 3.1(b) shows the logarithm of the distortion for the same frame. One can see that the R-D curves depend significantly on Q_{j-1}. The parameters of the independent models would thus depend on Q_{j-1}.

A second set of experiments is conducted. Several R-D curves are plotted in Figure 3.2 as a function of Q_{j-1}, the QP of the reference frame, considering several values of the QP of the P frame Q_j. In these curves, the reference frame is an I frame. The value of Q_{j-1} for the reference frame impacts significantly the R-D curve of the second frame. The larger the difference between the values of Q_{j-1} and Q_j, the larger is the impact on the R-D curve.



(a) (b)

Figure 3.1: Rate model (a) and Distortion model (b) using different QPs at the reference frame in Foreman sequence.



(a) (b)

Figure 3.2: The rate (a) and the logarithm of the distortion (b) of the j-th frame as a function of Q_{j-1}, parameterized with Q_j in Foreman sequence.

Dependent R-D models have thus to be introduced. In order to propose dependent R-D models, several regimes on the experimental curves in Figures 3.2 (a) and (b) are identified.

2.1 Rate regimes

All rate curves of Figure 3.2(a) decrease when Q_{j-1} decreases. When the quality of the reference frame increases, the motion compensation is usually more efficient, and the texture (motion compensation residuals) contains less energy, and requires thus less bits to be represented. Besides this tendency, three regimes may be identified for the rate curves.

When $Q_{j-1} \leqslant Q_j$, the rate remains approximately constant. As long as the quality of the reference frame remains good, it allows to get a texture containing mainly elements which cannot be efficiently predicted from the reference frame,

whatever its quality. The rate is thus mainly determined by the amount of elements which are difficult to predict in the P frame.

When Q_{j-1} is much larger than Q_j ($Q_{j-1} \geqslant Q_j + K$), the rate remains approximately constant. In this regime, the reference frame has been much coarsely quantized, and is thus of less quality. The motion compensation is not efficient and the amount of information in the texture is not significantly affected by a poor or very poor reference frame. The rate is thus essentially determined by Q_j.

The intermediate regime, when $Q_{j-1} \in [Q_j, Q_j + K]$, corresponds to the situation described at the beginning of this section. Increasing the quality of the reference frame reduces the amount of information of the texture, and thus the rate.

2.2 Distortion regimes

All distortion curves of Figure 3.2(b) decreases when Q_{j-1} decreases, as for the rate curves. The interpretation is the same: having a better reference increases the efficiency of the motion-compensation, and reduces the amount of information in the texture. Even if the texture is coarsely quantized, the reconstructed frame may be of good quality thanks to the good quality of the reference frame. Now two regimes may be observed.

When $Q_{j-1} \geqslant Q_j$, the distortion remains approximately constant, since it is essentially determined by Q_j.

When $Q_{j-1} \leqslant Q_j$, the distortion decreases when Q_{j-1} decreases. In this regime, the motion-compensation residuals of frame j are coarsely quantized compared to frame $j-1$. When the motion-compensation is performed, many blocks are well predicted, and for those blocks, the motion-compensation residuals are negligible when compared to the quantization step corresponding to Q_j. Many blocks have a distortion that is thus determined by Q_{j-1}.

The fact that the R-D curves do not converge to the same point when Q_{j-1} is small is due to the fact that independently of the value of Q_{j-1}, some block are not well predicted during motion-compensation. The corresponding texture for those blocks are thus of quite high energy. The R-D characteristics for those blocks are actually determined by Q_j, and have a behavior similar to I frames.

In the next Section, an analytical approach is investigated, considering a model in the transform domain to tackle the dependency between the DCT coefficients of blocks of the P frame and of its reference frame.

3 Independent regime-switching auto-regressive model

In this section, we propose an analytical dependent R-D model that explains the different rate and distortion regimes analyzed in Section 2.

3.1 Main ideas

In a video coder involving block-based motion compensation such as H.264/AVC, to encode the j-th frame F_j of a video sequence, previously encoded frames $\widetilde{F}_k$, $k < j$, belonging to the same GoP are used to build a prediction $\widehat{F}_j$ of F_j via motion

estimation and compensation. The prediction residual $F_j - \widehat{F}_j$ is then transformed, quantized, and entropy-coded. The reconstructed frame is then $\widetilde{F}_j$.

Neglecting the quantization noise, when the motion estimation and compensation is efficient, many collocated pixels in F_j and in $\widehat{F}_j$ are very similar. But, for some parts of the frame, the motion estimation may be less efficient, due, *e.g.*, to scene change, to appearing objects, or to motion of the camera. Collocated pixels in such areas may be different. Similar observations may be done when considering transformed coefficients (TC). The transform DCT coefficients can be classified into DC and AC coefficients. The DC coefficient is the mean value of the image block and carries most of the energy in the image block. The AC coefficients carry energy depending on the amount of detail in the image block. Most of the energy is compacted in the DC coefficient and a few AC coefficients. Collocated TC (DC and AC) may be similar or different depending on the quality of the motion estimation/compensation.

The effect of quantization noise is usually to decrease the efficiency of motion compensation. A prediction $\widehat{F}_j$ based on $\widetilde{F}_k$, $k \neq j$ is usually much less efficient than a prediction based on F_k, $k \neq j$. Our aim is to study the impact of the quantization noise of one frame on the next ones.

3.2 Definition

The dependency between collocated TC (or collocated pixels) is modeled here using an Independent Regime-Switching Auto-Regressive (IRS AR) model

$$Y_j = a_{X_j} Y_{j-1} + b_{X_j} U_j, \tag{3.1}$$

where Y_j and Y_{j-1} represent some collocated TC of a given block in frames j and $j-1$, X_j is a sequence of independent and identically distributed (iid) binary-valued random variables with $\Pr(X_j = 0) = 1-\rho$ and $\Pr(X_j = 1) = \rho$, and U_j is a sequence of iid zero-mean and unit variance Gaussian variables. The sequences X_j and U_j are assumed to be independent. This model allows taking into account the fact that in most cases (when $X_j = 0$), collocated TC are quite similar, in which case a_0 is close to one, and that sometimes (when $X_j = 1$), there is fewer correlation, in which case a_1 is closer to zero ($a_1 = 0$ in what follows).

Y_j is assumed wide-sense stationary. This imposes some constraints on the parameters a_0, b_0, and b_1. With $X_j = 0$, one gets

$$\sigma_y^2 \;=\; a_0^2 \sigma_y^2 + b_0^2 \sigma_u^2 = b_0^2 \sigma_u^2 / (1 - a_0^2) \tag{3.2}$$

and if $X_j = 1$, one obtains

$$\sigma_y^2 = b_1^2 \sigma_u^2. \tag{3.3}$$

Combining (3.2) and (3.3), one gets

$$b_1^2 = b_0^2 / (1 - a_0^2). \tag{3.4}$$

The R-D characteristics of the proposed model when scalar quantized with two different steps for Y_{j-1} and Y_j is studied in the following section.

3.3 Rate and distortion characteristics

Assume that Y_{j-1} is quantized with a scalar uniform quantizer q_{j-1} with step size Δ_{j-1}. For Y_j a predictive coding is performed with

$$\widehat{Y}_j = a_{X_j} q_{j-1}(Y_{j-1}), \tag{3.5}$$

as prediction for Y_j (the value of X_j is assumed to be known). The prediction residual

$$E_j^{X_j} = Y_j - \widehat{Y}_j = a_{X_j}\left(Y_{j-1} - q_{j-1}\left(Y_{j-1}\right)\right) + b_{X_j} U_j \tag{3.6}$$

is then quantized with a step-size Δ_j. Depending on X_j,(3.6) may become

$$E_j^0 = a_0\left(Y_{j-1} - q_{j-1}\left(Y_{j-1}\right)\right) + b_0 U_j, \tag{3.7}$$

when $X_j = 0$ corresponding to inter-coding and

$$E_j^1 = b_1 U_j, \tag{3.8}$$

when $X_j = 1$ corresponding to intra-coding. The aim of the remainder of this section is to provide R-D curves for the model in (3.1) as a function of Δ_{j-1} and Δ_j. In the H.264/AVC standard [II03], characteristics of the quantizers depend on a QP Q. The relation between the quantization step size Δ and the QP Q may be approximated as

$$\Delta\left(Q\right) = 2^{\frac{Q-4}{6}} / P_{\mathrm{F}}, \tag{3.9}$$

where P_{F} is a constant which value depends of the subband, see [MGL05]. The distortion for Y_j may be written as

$$D_j^{X_j}(\Delta_{j-1}, \Delta_j) = \int_{-\infty}^{+\infty} \left(x - q_j(x)\right)^2 f_{E_j^{X_j}}(x)\, dx, \tag{3.10}$$

where $f_{E_j^{X_j}}(x)$ is the probability density function of $E_j^{X_j}$. The rate required to represent the quantized Y_j is evaluated as the entropy of the output of the quantizer fed with $E_j^{X_j}$

$$R_j^{X_j}(\Delta_{j-1}, \Delta_j) = -\sum_{k=-\infty}^{+\infty} P_k(\Delta_{j-1}, \Delta_j) \log(P_k(\Delta_{j-1}, \Delta_j)) \tag{3.11}$$

where

$$P_k(\Delta_{j-1}, \Delta_j) = \int_{(k-\frac{1}{2})\Delta_j}^{(k+\frac{1}{2})\Delta_j} f_{E_j^{X_j}}(x)\, dx. \tag{3.12}$$

When $X_j = 1$, E_j^1 is zero-mean Gaussian with variance $\sigma_y^2 = b_1^2 \sigma_u^2$ and does not depend on Δ_{j-1},

$$f_{E_j^1}(x) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{x^2}{2\sigma_y^2}\right). \tag{3.13}$$

High-rate approximations for (3.10) and (3.11) are easily obtained. At medium to low rates (large values of Δ_j compared to σ_u), such high-rate approximation becomes coarse.

When $X_j = 0$, the probability density function of E_j^0 is

$$f_{E_j^0}(x) = \frac{1}{\sqrt{8\pi\sigma_y^2}} \sum_{k=-\infty}^{+\infty} \exp\left(-\frac{(x + k\Delta_{j-1}a_0)^2}{2\sigma_y^2}\right) G(x, \Delta_{j-1}, k), \tag{3.14}$$

where

$$\begin{aligned} G(x, \Delta_{j-1}, k) &= \operatorname{erf}\left(\frac{2a_0x + \Delta_{j-1}\left(1 - 2k\left(1 - a_0^2\right)\right)}{2\sqrt{2}\sigma_y\sqrt{1 - a_0^2}}\right) \\ &\quad - \operatorname{erf}\left(\frac{2a_0x - \Delta_{j-1}\left(1 + 2k\left(1 - a_0^2\right)\right)}{2\sqrt{2}\sigma_y\sqrt{1 - a_0^2}}\right). \end{aligned} \tag{3.15}$$

See details in Appendix A.

Since $f_{E_j^0}(x)$ and $f_{E_j^1}(x)$ are known, one may evaluate numerically $D_j^{X_j}(\Delta_{j-1}, \Delta_j)$ and $R_j^{X_j}(\Delta_{j-1}, \Delta_j)$ using (3.10) and (3.11). The expectation of the rate and the distortion for a TC with respect to X_j is then

$$D_j(\Delta_{j-1}, \Delta_j) = (1 - \rho)\, D_j^0(\Delta_{j-1}, \Delta_j) + \rho D_j^1(\Delta_j) \tag{3.16}$$

$$R_j(\Delta_{j-1}, \Delta_j) = (1 - \rho)\, R_j^0(\Delta_{j-1}, \Delta_j) + \rho R_j^1(\Delta_j), \tag{3.17}$$

since D_j^1 and R_j^1 do not depend on Δ_{j-1}.

3.4 Compound model

Assuming that the size of the transform is the same for the whole frame, N_{DC} blocks of TC have to be considered, each of which containing a single DC coefficient and N_{AC} AC coefficients. Assume that each of these coefficients is represented by our IRS-AR model, and that the quantization steps do not change within a frame, one gets a total distortion and rate for the j-th frame expressed as follows

$$\begin{aligned} D_j(\Delta_{j-1}, \Delta_j) &= \tfrac{1}{N_{\mathrm{DC}}(1+N_{\mathrm{AC}})} \textstyle\sum_{n=1}^{N_{\mathrm{DC}}} \left(D_j^{\mathrm{DC},n} + \sum_{\ell=1}^{N_{\mathrm{AC}}} D_j^{\mathrm{AC},n,\ell}\right), \\ R_j(\Delta_{j-1}, \Delta_j) &= \textstyle\sum_{n=1}^{N_{\mathrm{DC}}} \left(R_j^{\mathrm{DC},n} + \sum_{\ell=1}^{N_{\mathrm{AC}}} R_j^{\mathrm{AC},n,\ell}\right), \end{aligned} \tag{3.18}$$

The number of parameters of the resulting compound model would be prohibitively large. Several simplifications have thus to be considered. First, it is assumed that when for a given TC block, the model for the DC and AC coefficients are switching simultaneously. This is reasonable, since when for a block, a motion compensation is not efficient, there is energy at all frequencies. Second, all IRS-AR models for the DC coefficients are aggregated within a single *average* IRS-AR model. Similarly, N_{AC} *averaged* IRS-AR models for the AC coefficients may be considered (one per frequency). This would lead to $1 + N_{\mathrm{AC}}$ models, which still leads to a large number of parameters. Usually, most of the energy is gathered in the low-frequency AC coefficients. One thus represents the R-D behavior of the AC coefficients by a single IRS-AR model which is quite coarse approximation.

As a result, one obtains a compound model consisting of two IRS-AR models (one for the DC coefficients and one for all AC coefficients)

$$Y_j^{\mathrm{DC}} = a_{X_j}^{\mathrm{DC}} Y_{j-1}^{\mathrm{DC}} + b_{X_j}^{\mathrm{DC}} U_j, \tag{3.19}$$

and

$$Y_j^{\mathrm{AC}} = a_{X_j}^{\mathrm{AC}} Y_{j-1}^{\mathrm{AC}} + b_{X_j}^{\mathrm{AC}} U_j. \tag{3.20}$$

The expression of the rate and distortion becomes then

$$R_j = \kappa(\delta R_j^{\mathrm{DC}} + (1-\delta) R_j^{\mathrm{AC}}) \tag{3.21}$$

end

$$D_j = \delta D_j^{\mathrm{DC}} + (1-\delta) D_j^{\mathrm{AC}}. \tag{3.22}$$

A multiplicative constant δ is introduced to weight the contributions of DC and AC coefficients to the total rate and distortion. An additional parameter κ is introduced in the rate to take into account the number of blocks of the frame. One may take $\kappa = N_{\mathrm{DC}}$, but introducing an additional degree of freedom in the model of the rate helps mitigating all approximations which were previously considered.

A least square estimation of the parameter vector $\mathbf{p} = (a_0^{DC}, a_0^{AC}, b_0^{DC}, b_0^{AC}, \sigma_u^{DC}, \sigma_u^{AC}, \delta, \rho)$ is as follows

$$\widehat{\mathbf{p}} = \arg\min \sum_{k=1}^{N} (D_j(\Delta_{j-1}^k, \Delta_j^k) - D_j^{\exp}(\Delta_{j-1}^k, \Delta_j^k))^2. \tag{3.23}$$

Here D_j is the distortion calculated using (3.10), (3.16), and (3.22) at frame j, whereas $D_j^{\exp}$ is the experimental distortion using H.264/AVC at the same frame j with N values of the pair of quantization steps $(\Delta_{j-1}^k, \Delta_j^k)$ or equivalently of the QPs (Q_{j-1}^k, Q_j^k), $k = 1, \ldots, N$. The parameter κ is then adjusted using a least square estimation with the same experimental points on the rate curve, but with $\mathbf{p}$ fixed at $\widehat{\mathbf{p}}$.

3.5 Experimental results

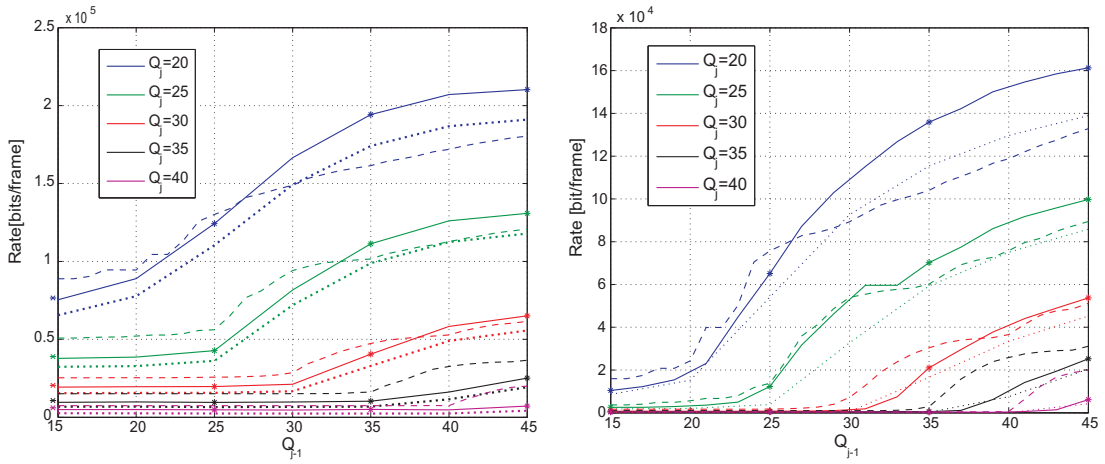
Experimental tests on the rate and the distortion evolution of the reference and the predicted frames of Soccer and Container are considered. GoP structure is considered as IPPP... in both sequences. Experimental characteristics are obtained with the H.264/AVC encoder.

A subset of $N = 20$ measurements are considered in (3.23) to fit the parameter of the proposed compound model using the following QP couples $(Q_{j-1}, Q_j) = ((25,15),(35,15),(45,15),(25,20),(35,20),(45,20),(25,25),(35,25),(45,25),(25,30),(35,30),(45,30),(25,35),(35,35),(45,35),(25,40),(35,40),(45,40))$. The obtained rate and distortion characteristics are compared to the experimental ones.

Figure 3.3 represents the evolution of the rate for the second frame (Inter-coded) of Soccer and Container sequences respectively in QCIF format while varying the QP of the first reference (Intra-coded) frame. Figure 3.4 represents the logarithm of the corresponding distortion.

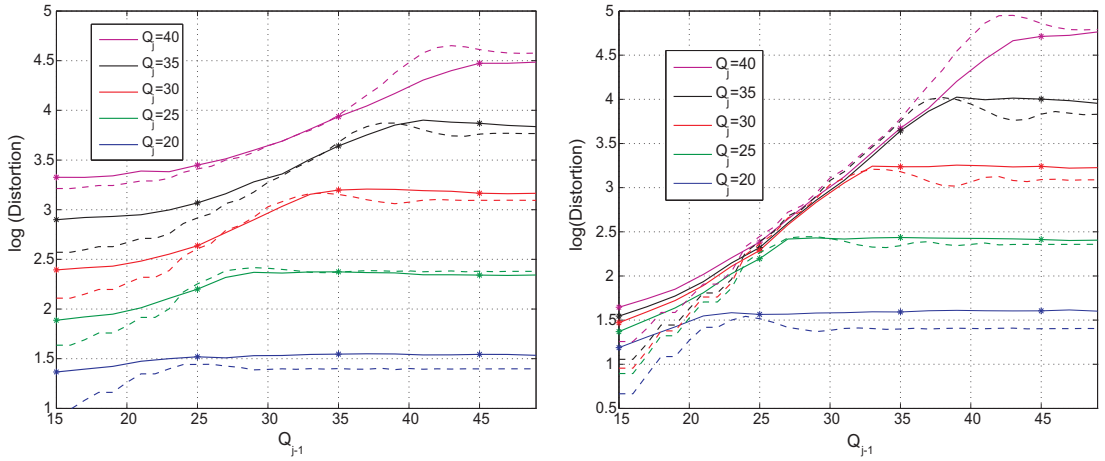
The estimated parameters values resulting from the resolution of equation (3.23) and used in R-D curves in Figures 3.3(a) and 3.4(a) on the first P-frame in Soccer sequence are: $a_0^{DC} = 0.53$, $a_0^{AC} = 0.9$, $b_0^{DC} = b_0^{AC} = 1$, $\sigma_u^{DC} = 3.09$, $\sigma_u^{AC} = 7.8$, $\delta = 0.73$, $\rho = 0.06$, and $\kappa = 8500$.

The estimated parameters values resulting from the resolution of equation (3.23) and used in R-D curves in Figures 3.3(b) and 3.4(b) on the first P-frame in Container



(a) (b)

Figure 3.3: Rate as a function of Q_{j-1}, parameterized with Q_j; compound model (dashed), total rate using H.264/AVC (solid), texture rate using H.264/AVC (dotted) for Soccer(a) and Container(b) sequences.



(a) (b)

Figure 3.4: Log of the distortion as a function of Q_{j-1}, parameterized with Q_j; compound model (dashed), H.264/AVC (solid) for Soccer(a) and Container(b) sequences.

sequence are: $a_0^{DC} = 88$, $a_0^{AC} = 0.93$, $b_0^{DC} = b_0^{AC} = 1$, $\sigma_u^{DC} = 1.84$, $\sigma_u^{AC} = 7.28$, $\delta = 0.73$, $\rho = 0.01$, and $\kappa = 6800$.

Figure 3.4 shows that the compound model is able to describe quite accurately the evolution of the distortion (for the luminance) and its various regimes discussed in section 2. Figure 3.3 illustrates the evolution of the rate as a function of Q_{j-1}, parameterized with Q_j; compound model (dashed), total rate using H.264/AVC (solid), texture rate using H.264/AVC (dotted) for Soccer(a) and Container(b) sequences. Dots on the solid curves correspond to the experimental values used for the parameter estimation as defined in (3.23). Here the fit with the texture part of the rate is satisfying, except for large values of the QP. This is mainly due to the fact that the dependency between DC coefficients has not been taken into account in the proposed compound model. When large values of QP are considered, DC

values get closer, and are more efficiently entropy-coded. This is why the rate is overestimated. In addition, the rate due to signaling and transmission of motion vectors is not taken into account. Additional adjustments have to be performed to account for the cost of motion vectors and signalization to reach an accurate fit of the total rate using H.264.

4 Conclusion

In order to process rate control at the frame level, dependent R-D model are needed. The inter-frame dependency is interpreted theoretically by considering the quantization of an independent regime-switching auto-regressive source model with varying QPs. By comparing the R-D performance between the analytical model and the real H.264/AVC curves, we have shown the accuracy of the proposed model to describe the H264 encoder behavior. Experimental results using our proposed dependent R-D model show that the proposed compound model is able to accurately represent the distortion dependency between frames. For the rate, the fit with the texture part of the rate is satisfying, except for large values of the quantization parameter. This is mainly due to the fact that the dependency between DC coefficients has not been taken into account in the proposed compound model.

A dependent model can be very usefully for control systems performing rate control at the frame level. Such systems require information about the impact of a chosen encoding parameter in a reference frame on the R-D values of the frame that will be predicted from it. But, as we have seen in the previously proposed dependent R-D models and in our proposed IRS-AR model, much more model parameters are involved in the R-D models than in the independent model. Thus more encoding trials are needed, increasing the control complexity. In the next chapter we propose a rate control system that regulates the rate and the distortion at the GoP level. Thus only independent models are used.

Part II

Statistical Multiplexing: Deterministic Optimization

Chapter 4

Video Statistical Multiplexing over Broadcast Networks

1 Introduction

In this chapter, we propose a statistical multiplexing (SM) system of video contents over broadcast channels. SM of video contents aims at transmitting several variable bit rate (VBR) encoded video streams over a band-limited channel. Buffering at the output of encoders is one of the several techniques used to smooth out the fluctuating bit rate of compressed video due to variations in the activity of video contents. A closed-loop control of both video encoders and buffers is performed jointly. First, a predictive joint video encoder controller accounting for minimum quality, fairness, and smoothness constraints is considered. Second, all buffers are controlled simultaneously to regulate the buffer bit level first, and the buffer delays second. The buffer bit level and the delay are adjusted according to a reference buffer bit level and a reference delay constraints. The main idea is to update the encoding rate for each video unit according to the average level of the buffers, to maximize the quality of each program and effectively use the available channel rate. The predictive joint video encoder controller accounting for minimum quality, fairness, and smoothness constraints has led to the submission of a patent application [CSK09b] and was published in the IEEE Packet video Workshop [CSK09c]. The work on the closed-loop control of both video encoders and buffers has led to the submission of a patent application [CSK09a], then to a publication in the IEEE Globecom conference paper [CSK10b], and to the submission of a journal paper in the IEEE transaction on broadcasting.

1.1 Problem statement

Due to limited bandwidth resources, video programs are compressed using efficient video encoders such as MPEG 4 [Koe02], H.264/AVC [WSBL03], or H.264/SVC [SMW07]. The compressed programs are then multiplexed with other contents. Two encoding modes may be considered, leading to two types of multiplexing. Constant Bit Rate (CBR) encoding leads to an equal distribution of the channel rate between programs without any consideration about their respective complexity. This scheme is simple, but the quality may vary significantly with time within a single program

and between programs. Encoding with Variable Bit Rate (VBR) [CGJ06] allows a simpler program to be encoded with a low rate leaving remaining rate to other programs with more complex contents, *e.g.*, action motion pictures. The purpose of SM [Cha02] is then to share efficiently the channel rate between programs via a dynamic adjustment of the coding rate of each program.

Apart from the optimal use of the available channel rate, SM systems may target the satisfaction of several constraints linked to the quality-of-service (QoS) of the delivered programs, including video quality and transmission delay. For example, SM systems may be designed in such a way that

- programs are encoded with a minimum quality (*minimum quality* constraint [VTT07]),
- programs are encoded with similar quality (*fairness* constraint [JBT+08]),
- the quality of each decoded program varies smoothly with time (*smoothness* constraint [TVT08]),
- buffers bit levels are controlled to prevent buffer under and overflow [MMS03].
- latency at the receiver side, including switching between programs, is minimized [RBG09].

Finding an SM system able to satisfy simultaneously all these constraints in the context of video broadcasting is still a challenging task. This is mainly due to the non-stationary content of each program. Variations may be due, *e.g.*, to scene change or to high activity within a program. We propose an SM system that allows transmission of several video programs over a broadcast channel while taking into account the minimum quality, the fairness, and the smoothness constraints. Buffer control is also addressed by considering separately both buffer bit level control and buffer delay control cases.

1.2 Related works

The availability of well-tuned Rate-Distortion (R-D) or complexity models for each program is very useful to satisfy the previously-mentioned constraints. These models can be obtained using the *feedback* approach, where R-D statistics generated by the encoder are used to control the future bit-rate allocation, or using the *look-ahead* approach, where R-D statistics are used to adjust the bit rate *prior* to coding the frames in question. These two approaches are discussed in [PA95]. Some R-D models are recalled in Chapter 2.

The R-D trade-off of encoded streams may be adjusted by selectively discarding frames as in [ZNAT01, LZK+09] or via the encoding parameters as in [MGL05, HW08]. In the case of scalable video encoders, *e.g.*, H.264/SVC, this control may be replaced by some packet filtering process [SVHR10, MK10]. In this case, the number of transmitted enhancement layers for each frame is the control parameter.

Once the R-D characteristics are available, one can control the encoding parameters (QP, number of skipped frames, inter-frame prediction parameter, *etc.*), using a rate control algorithm. Various algorithms have been proposed in the context of

single and multiple video encoding. The rate control problem for single video is addressed in [LO98] by comparing the quality of previously encoded frames to that of the current frame in order to interpolate the R-D characteristics and to determine the appropriate encoding parameters. However, this technique accounts only for the past and may lead to the violation of quality constraints. In [RBG08], a joint encoder and statistical multiplexer of video programs is proposed. The proposed control system allows decreasing the end-to-end delay and improving the average quality of compressed video by dynamically distributing the available bit rate between the video sources according to their relative complexity. A smooth video quality is achieved by allowing only small variations of the current frame quality compared to the average PSNR of previously encoded frames and by using a low-pass filter to smooth the QP variations. This method may lead, however, to difficulties in case of a scene change or high motion in the video programs.

To handle the complicated inter-frame dependency problem, the R-D characteristics of the frames next to the one that has to be encoded are exploited in [HW08]. This look-ahead approach in an SM context allows getting the R-D characteristics of future uncoded video frames within the look-ahead window with a moderate computational complexity. Good smoothness over frames and fairness are obtained with the proposed control system, however, these two constraints are not considered explicitly in the optimization problem which makes them difficult to be achieved with other system conditions (channel rate, scene change, *etc*).

Among all constraints, the smoothness is the most difficult to satisfy due to the non-stationary content of video programs. This constraint has been considered in the context of single-source video coding, *e.g.*, in [LXLZ01,MAvdS+03,KLLZ06], and in an SM context *e.g.*, in [VTT07, TVT08]. The SM system proposed in [TVT08] aims at minimizing the variance of the distortion of the encoded frames. This allows getting a better video quality in average. The encoding rate is adjusted so that the rate constraint is updated according to the level of a shared buffer achieving the target smoothness distortion constraint. However, such shared buffer hinders the control of the buffer delay for each individual multiplexed program.

The buffer control is another important issue to ensure a good use of the available channel rate and to limit the video delivery delay. Buffer control has been considered, *e.g.*, in [BNW99,ZRJ91,MMS03,WCSQ97]. For example, in [MMS03], several streams are multiplexed and their transmission rates are adapted using buffer level information. This method allows a simultaneous adjustment of the transmission rate and the buffer level, however, the buffer occupancy is controlled at a bit level which does not allow to control the buffering delay. The rate allocation method proposed in [WZ09] minimizes the allocated resource while satisfying some QoS requirements. QoS is related to both buffer level in bits and the buffer delay (probability of buffer overflow and delay violation). Rate control and buffer delay are also addressed in [RBG09] where a technique for performing SM in conjunction with time slicing in DVB-H, implemented in the IP encapsulator is proposed. This method achieves a satisfactory use of the channel rate and a minimum buffer delay for all multiplexed programs. Nevertheless, in both [RBG09] and [WZ09], no constraint on the video quality is considered.

A summary of previous results concerning SM systems with the considered constraints is provided in Table 4.1 where (X) means that the constraint is experimen-

tally achieved with the considered scenario but not explicitly targeted.

SM	Constraints				Buffer control		Long term	Solution
	Channel rate	Min quality	Fairness	Smooth-ness	Bits	VU	control	
[RBG09]	X				X			Analytical
[WZ09]	X				X	X		NLO
[EM94]	X				X			Stochastic Opt.
[VTT07, TVT08]	X		(X)	X	X			ALO
[WV96]	X		(X)	(X)	X			Analytical
[VH08]	X			(X)				Numerical
[RBG08]	X		(X)	X	X			Analytical
[HW08]	X		(X)	(X)	X		X	Numerical
Here	X	X	X	X	X	X	X	NLO and ALO

Table 4.1: Summary of the properties of various SM systems, NLO (Lagrange optimization with numerical solution), ALO (Lagrange optimization with analytic solution).

1.3 Main contributions

We introduce an SM system which performs a joint closed-loop control of video encoders and buffers. The control is performed at a video unit (VU) level to provide at any time instant a smooth quality between the VUs of a given program and bounded quality differences between the VUs of the multiplexed programs. VU may be a frame or a GoP. In addition to quality constraints, the channel rate constraint as well as buffer constraint are also targeted. The buffer management is addressed in two different ways: First, the bit levels of all buffers are controlled simultaneously to prevent buffer under and overflow; Second, a similar transmission delay for all programs is targeted.

The parameters of all video encoders are adjusted for each VU using a *predictive control* over a window containing the previous, current, and several future VUs. This technique allows a better satisfaction of the quality constraints compared to non-predictive control. For that purpose, the R-D characteristics of the current and future VUs have to be estimated.

The bit levels of the buffers are adjusted via the transmission rate of each program to fully use the available channel rate and control the bit level and the buffer delay. When the buffer delay constraint is considered, the control is performed at the VU level. Thus, delivery and program switching delay are thus better controlled. The closed-loop is obtained using a Proportional-Integral-Derivative (PID) feedback of the buffer level information to the controller of the video encoders, which allows the encoding rate constraint to be dynamically updated.

2 Architecture of the proposed statistical multiplexing system

Figure 4.1 presents the proposed architecture to perform SM of N video programs encoded and transmitted in parallel over a broadcast channel; no feedback is considered. In this context no receiver feedback is considered, and the level of the buffers at receiver side is not available.

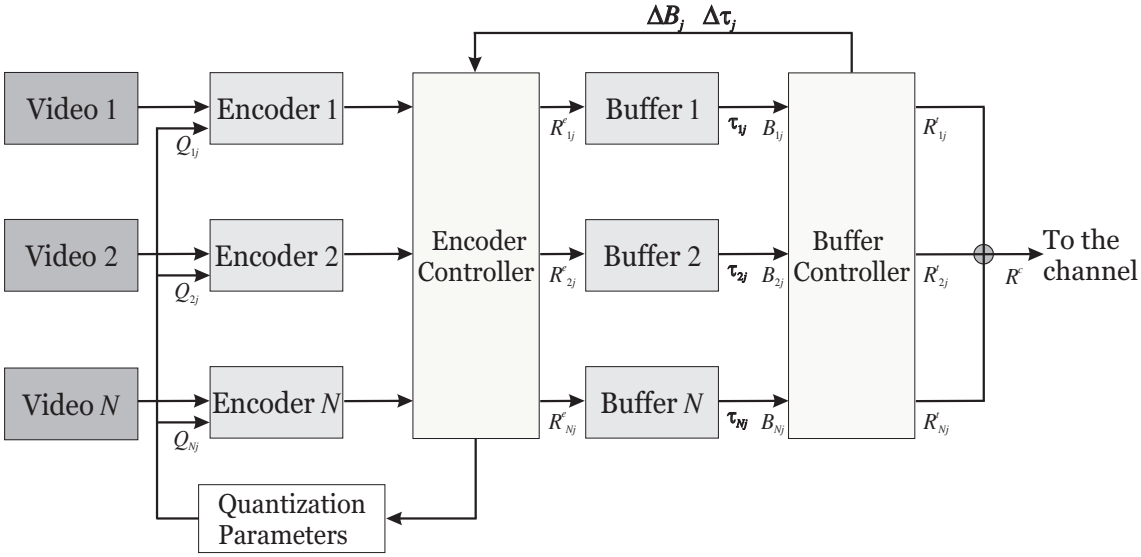


Figure 4.1: Proposed closed-loop statistical multiplexing system.

V_{ij} is the j-th VU in the i-th video program. All VUs are assumed to have the same duration T. The frame rate F as well as the number of frames per VU N_f are assumed constant with time and identical for all multiplexed programs. The QP Q_{ij} is one of the parameters used by each encoder to compress V_{ij}.

The regulation process provides one QP per VU, which is then fed to the video encoder. When the VU is considered as a Group of Pictures (GoP), the encoder may use the same QP for each frame in the GoP, which provides a more or less constant quality. It may also perform an adjustment of QP for each frame around the provided value to perform a R-D optimization, using some rate control algorithm, see *e.g.*, [KLLZ06].

Between time $j-1$ and j, denoted by time instant j, the encoder controller determines Q_{ij} so that the encoding rates R^{e}_{ij}, $i = 1 \ldots N$ of the N encoders satisfy some dynamically updated rate constraint R_j, defined later, while satisfying the quality constraints. At each encoder output, the i-th buffer stores temporarily N^{VU}_{ij} encoded VUs. Both buffer bit level B_{ij} and buffer delay τ_{ij} can be controlled. The transmission rate R^{t}_{ij} is allocated to all buffers $i \in \{1 \ldots N\}$ so that the channel rate constraint R^{c}_j is satisfied. For that purpose, in the first case, a reference buffer bit level is defined and denoted by B_0, expressed in bits, corresponding to the reference targeted value for all buffers bit levels. In the second case, a reference delay denoted by τ_0, expressed in seconds corresponds to the reference targeted buffer delay. The differences $B_{ij} - B_0$ averaged over the N programs is denoted by ΔB_j. The differences $\tau_{ij} - \tau_0$ averaged over the N programs is denoted by $\Delta\tau_j$. ΔB_j

Notation	Definition	Notation	Definition
N	Number of multiplexed programs	S_j	Channel state at time j
N_{f}	Number of frames per VU	N_{ij}^{VU}	Number of VU in buffer i at time j
T	Video unit duration	B_{ij}	Bit level of the i-th buffer at time j
i	Index of the video program	τ_{ij}	Buffer delay in the i-th buffer at time j
j	Time index	B_0	Reference buffer bit level
V_{ij}	Video unit of the i-th program at time j	τ_0	Reference delay
F_i	Frame rate of program i	ΔB_j	Average buffer bit level deviation among programs at time j
Q_{ij}	Quantization parameter of the i-th program at time j	$\Delta\tau_j$	Average delay deviation among programs at time j
R_{ij}^{e}	Encoding rate of the i-th program at time j	$\Delta P_{ij}^{\mathrm{s}}$	Smoothness constraint
P_{ij}	PSNR of the i-th program at time j	$\Delta P_{ij}^{\mathrm{f}}$	Fairness constraint
R_{ij}^{t}	Transmission rate of the i-th program at time j	K_{P}	Proportional gain in the control system
R_j^{c}	Channel rate at time j	K_{I}	Integral gain in the control system
R_j	Encoding rate constraint at time j	K_{D}	Derivative gain in the control system

Table 4.2: Notations.

and $\Delta\tau_j$ are fed back to the encoder controller that use it to get the rate constraint R_{j+1} for the next VU. Buffers and video encoders are thus controlled in a closed loop, see Figure 4.1.

To control the video quality, several objective and subjective video quality measurement techniques are available, see, *e.g.*, [ITU08,SW98,Win05] and the references therein. Here, as in most of existing works [HW08, SVHR10, VH08], we use the Peak-Signal-to-Noise Ratio (PSNR) metric. The PSNR of the j-th frame in the i-th program is $P_{ij} = 10 \log_{10} \left(\frac{255^2}{D_{ij}} \right)$ where D_{ij} is the distortion (considering a quadratic distortion measure).

Table 4.2 summarizes the notations used in this chapter.

3 Optimization problem formulation

The aim of the proposed SM system is at each time instant j to provide quantization parameters Q_{ij} to the video encoders and transmission rate R^{t}_{ij} to the buffers for all programs, while satisfying some QoS constraints. Due to variations of the R-D characteristics of video contents, some short-term decisions taken at time j, considering only the state of the system at time j, may lead to violations of the constraints at some time instant $j' > j$.

The solution proposed for this problem is to perform the control of the video coders over a time window of W VUs for each program, from VU $j-1$ to VU $j+W-2$, see Figure 4.2. At time j and for each program i, the encoder controller evaluates a vector $\mathbf{Q}_i^{(j)} = \left(Q_{ij}^{(j)} \dots Q_{ij+W-2}^{(j)} \right)$. Only the parameter $Q_{ij}^{(j)}$ evaluated for VU j is applied at time j, the parameters $Q_{ij+k}^{(j)}$ evaluated for future VUs, $k = 1 \dots W-2$, are not applied but updated at the next time steps. This foresighted control allows choosing a value for the control parameters $\mathbf{Q}_i^{(j)}$ that satisfies the constraints at time j and for which one knows that there exists values $Q_{ij+k}^{(j)}$ such that the constraints are also satisfied at the future time instants considered in the control window.

No predictive control is performed for the buffers, but their level is fed back to the video encoder controller, see Figure 4.1.

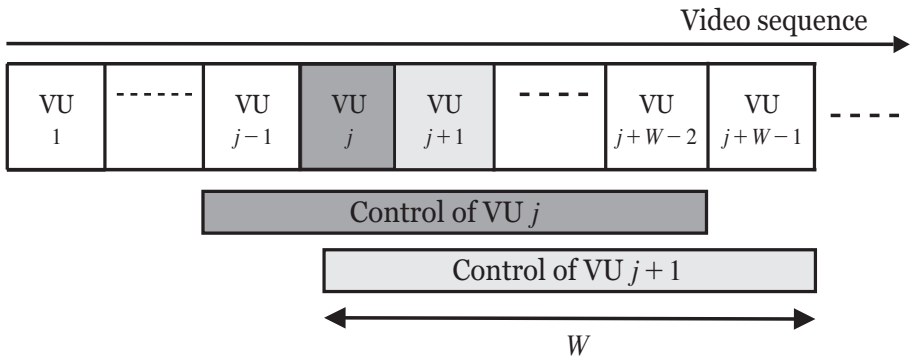


Figure 4.2: Predictive control involving W VUs.

As in [MGL05] and [YDXD05], we consider the following PSNR-Rate model

$$P(Q) = a_{\mathrm{P}} Q + b_{\mathrm{P}}, \tag{4.1}$$

$$R(Q) = a_{\mathrm{R}} \exp(-b_{\mathrm{R}} Q), \tag{4.2}$$

for the PSNR and the rate as a function of the QP Q due to its moderate complexity and its accuracy to fit the actual rate and the PSNR at the GoP level with only two encoding trials, see Section 3.1.2 in Chapter 2.

In the following sections, the joint encoder and buffer control problem is formulated as a constrained optimization problem.

3.1 Cost function

The aim of the proposed SM system is to maximize the average quality of the broadcasted video programs. The considered cost function

$$\left(\widehat{\mathbf{Q}}_1^{(j)} \dots \widehat{\mathbf{Q}}_N^{(j)}\right) = \arg \max_{\mathbf{Q}_1^{(j)} \dots \mathbf{Q}_N^{(j)}} \sum_{k=j}^{j+W-2} \gamma^{k-j} \sum_{i=1}^{N} P_{ik}\left(Q_{ik}^{(j)}\right) \tag{4.3}$$

allows performing the maximization of a weighted sum of PSNRs over the control window of W VUs. The discount factor $0 < \gamma \leqslant 1$ provides more weight to the PSNR of current VUs for which the channel conditions and the buffer levels are well known, contrary to future PSNRs for which they are less precisely determined.

3.2 Rate constraints

Several rate constraints have to be satisfied, as introduced in Section 2.

3.2.1 Channel model

In the considered scenario, the bandwidth (and rate) allocated to the broadcast channel may vary with time, as in [CHC07]. These variations may be due to concurrent services, which may leave more or less resources to the broadcast service, readily considered *e.g.*, in the Long Term Evolution (LTE) standard [STB09].

The state of the broadcast channel is assumed to vary slowly with time and to be represented by a first-order Markov sequence $\{S_j\}$ [JKM+92, MGO04] with values between 1 and n corresponding to n channel rates $R^1 \dots R^n$. $S_j = k$ means that the channel rate between time j and $j+1$ is R^k. The state transition probabilities $p_{h,k} = p(S_j = h \,|\, S_{j-1} = k)$, as well as the rates are assumed known *a priori*, they may also be estimated on-line. When performing the control at time j, the realization of S_j is also assumed to be known.

3.2.2 Encoder rate constraints

In average, the sum of the encoding rates should be equal to the channel rate. Thanks to the buffers, some rate variations may be tolerated. In the proposed scheme, at time j, a dynamically updated encoder rate constraint R_j is provided by the buffer controller to the encoder controller, leading to

$$\sum_{i=1}^{N} R_{ij}^{\mathrm{e}}(Q_{ij}^{(j)}) = R_j. \tag{4.4}$$

Since it is rather difficult to accurately anticipate the future buffer levels, the encoder rate constraints for VUs between time $j+1$ and $j+W-2$ are taken as the

expected value of the channel rate, knowing the current channel state S_j. Thus, the following constraints are introduced when performing the predictive control at time j

$$\sum_{i=1}^{N} R^{\mathrm{e}}_{ij+k}(Q^{(j)}_{ij+k}) = E\left(R^{\mathrm{c}}_{j+k}|S_j\right), \tag{4.5}$$

for $k = 1 \ldots W-2$, with $R^{\mathrm{c}}_j = R^{S_j}$. Satisfying (4.5) requires at time j the availability of the R-D characteristics of VUs at time $j+k$, $k = 1 \ldots W-2$. This introduces a constant additional transmission delay $(W-2)\,T$ due to the buffer of $W-2$ *future* VUs.

3.2.3 Transmission rate and buffer bit level control

All buffers are controlled simultaneously to limit deviations from the reference buffer bit level B_0 to prevent buffer under and overflow. The level of each buffer is controlled by adjusting the output rate R^{t}_{ij}, solution of the following constrained optimization problem

$$\widehat{\mathbf{R}}^{\mathrm{t}}_j = \left(\widehat{R}^{\mathrm{t}}_{1j}, \ldots, \widehat{R}^{\mathrm{t}}_{Nj}\right) = \arg\min_{\mathbf{R}^{\mathrm{t}}_j} \sum_{i=1}^{N} (B_0 - B_{ij+1})^2, \tag{4.6}$$

subject to

$$\textstyle\sum_{i=1}^{N} R^{\mathrm{t}}_{ij} = R^{c}_{j}, \tag{4.7}$$

where

$$B_{ij+1} = B_{ij} + \left(R^{\mathrm{e}}_{ij} - R^{\mathrm{t}}_{ij}\right) T. \tag{4.8}$$

Here, we consider that the encoding rate R^{e}_{ij} and the transmission rate R^{t}_{ij} and provided respectively by the encoder controller and the buffer controller between time $j-1$ and time j at time j. The buffer state transition is assumed to be performed between time j and time $j+1$ using R^{e}_{ij} and R^{t}_{ij} values. An explicit expression for $\widehat{\mathbf{R}}^{\mathrm{t}}_j$ can be obtained. The constrained optimization problem in (4.6) and (4.7) can be solved using the Lagrange multiplier method.

$$\mathcal{L}(R^{\mathrm{t}}_{1j}, \ldots, R^{\mathrm{t}}_{Nj}, \lambda) = \sum_{i=1}^{N} (B_0 - B_{ij+1})^2 + \lambda \left(\sum_{i=1}^{N} R^{\mathrm{t}}_{ij} - R^{c}_{j}\right). \tag{4.9}$$

Using (4.8) in (4.9),

$$\mathcal{L}(R^{\mathrm{t}}_{1j}, \ldots, R^{\mathrm{t}}_{Nj}, \lambda) = \sum_{i=1}^{N} (B_0 - B_{ij} - \left(R^{\mathrm{e}}_{ij} + R^{\mathrm{t}}_{ij}\right) T)^2 + \lambda \left(\sum_{i=1}^{N} R^{\mathrm{t}}_{ij} - R^{\mathrm{c}}_{j}\right). \tag{4.10}$$

The derivative of (4.10) with respect to R^{t}_{ij} is

$$\frac{\partial \mathcal{L}(R^{\mathrm{t}}_{1j}, \ldots, R^{\mathrm{t}}_{Nj}, \lambda)}{\partial R^{\mathrm{t}}_{ij}} = \left[-2R^{\mathrm{t}}_{ij}T^2 - 2T\left(B_0 - B_{ij} - R^{\mathrm{e}}_{ij}T\right) + \lambda\right]. \tag{4.11}$$

From $\frac{\partial \mathcal{L}(R^{\mathrm{t}}_{1j}, \ldots, R^{\mathrm{t}}_{Nj}, \lambda)}{\partial R^{\mathrm{t}}_{ij}} = 0$, one can get the transmission rate

$$R^{\mathrm{t}}_{ij} = \frac{1}{T}\left(-\frac{\lambda}{2T} - B_0 + B_{ij} + R^{\mathrm{e}}_{ij}T\right). \tag{4.12}$$

From the rate constraint (4.7), one can obtain λ

$$\lambda = 2T \left[-R_j^{\mathrm{c}} \frac{T}{N} - B_0 + \frac{1}{N} \sum_{i=1}^{N} (B_{ij} + R_{ij}^{\mathrm{e}} T) \right]. \tag{4.13}$$

Thus, the transmission rate is expressed as follows

$$R_{ij}^{\mathrm{t}} = \frac{R_j^{\mathrm{c}}}{N} + \frac{1}{T} \left(B_{ij} + R_{ij}^{\mathrm{e}} T - \frac{1}{N} \sum_{i=1}^{N} (B_{ij} + R_{ij}^{\mathrm{e}} T) \right). \tag{4.14}$$

Introducing (4.14) in (4.8), one obtains the buffer bit level

$$B_{ij+1} = \frac{1}{N} \sum_{i=1}^{N} \left(B_{ij} + R_{ij}^{\mathrm{e}} T \right) - \frac{R_j^{\mathrm{c}} T}{N}, \tag{4.15}$$

showing that the control tends to equalize the bit level of all buffers.

In this case, the buffer level has to remain close to some reference delay B_0, which is chosen not too large to limit the global delivery delay, but not too small to mitigate the variations with time of the R-D characteristics of video programs and of the channel rate.

At time j, the average buffer level deviation ΔB_j of each buffer bit level B_{ij} from the reference bit level B_0 is

$$\Delta B_j = \frac{1}{N} \sum_{i=1}^{N} (B_{ij} - B_0). \tag{4.16}$$

When $\Delta B_j > 0$, the bit levels of the buffers are in average higher than the reference level B_0 and the encoding rate of the next VUs should be reduced. When $\Delta B_j < 0$, the buffers are draining too fast and the encoding rate may be temporarily increased. We propose here to evaluate an updated encoding rate constraint R_j at the buffer controller using a Proportional Integral Derivative (PID) control [SH05]

$$R_j = R_j^c - K_{\mathrm{P}} \frac{\Delta B_j}{T} - K_{\mathrm{I}} \sum_{k=1}^{j} \frac{\Delta B_k}{T} - K_{\mathrm{D}} \left(\frac{\Delta B_j}{T} - \frac{\Delta B_{j-1}}{T} \right), \tag{4.17}$$

and to feed it back to the encoder controller. In (4.17), K_P is the Proportional (P) gain, K_I the Integral (I) gain, and K_D is the Derivative (D) gain. It is well known that P control cannot eliminate steady-state error. Usually, the steady-state error decreases when K_P increases. However, a large K_P may lead to instability. The contribution from the I term is proportional to accumulated errors, and aims at canceling the steady-state error. The D term is used to reduce the magnitude of the overshoot produced by the I term and to improve the closed-loop stability. Various methods have been proposed to tune the PID parameters, see, *e.g.*, [vdZ08].

The control in (4.17), allows a regulation of the incoming flow by updating the encoding rate constraint. Such a regulation is similar to that used in the back-pressure mechanism [KK96]. We assume that the feedback signal (ΔB_j or R_j is available instantaneously at the encoder controller and used to select the appropriate QPs for the next VUs.

3.2.4 Transmission rate and buffer delay control

In this section, for each buffer, the transmission rates R^{t}_{ij} at time j are chosen to provide an equal buffer delay to the N programs and fully use the channel rate. The latter constraint leads to an average switching delay between programs (at least for what concerns the time to get a new Intra-coded frame) independent of the target program[1], and to better control the delivery delay than with a control of the buffer fullness in bits.

At time j, the transmission rates R^{t}_{ij} have thus to be such that

$$\sum_{i=1}^{N} R^{\mathrm{t}}_{ij} = R^{\mathrm{c}}_{j}, \tag{4.18}$$

and having equal delays among programs leads to

$$\tau_{ij+1} = \tau_{i'j+1}, \;\; i, i' = 1 \ldots N. \tag{4.19}$$

The system can be represented as an optimization problem as in Section 3.2.3 or it can be solved as simultaneous equations. The delay τ_{ij+1} in buffer i at time $j+1$ is difficult to determine accurately, since the buffers are drained bit-by-bit. Assuming that at time j, the bits of the encoded VU are regularly fed to the buffer with a rate R^{e}_{ij}, and that it is regularly drained with rate R^{t}_{ij}, the buffer level in bits B_{ij+1} at time $j+1$ is

$$B_{ij+1} = B_{ij} + \left(R^{\mathrm{e}}_{ij} - R^{\mathrm{t}}_{ij}\right) T. \tag{4.20}$$

One gets the following estimate of τ_{ij+1}

$$\tau_{ij+1} = \frac{B_{ij+1}}{\bar{R}^{\mathrm{e}}_{ij}}, \tag{4.21}$$

where $\bar{R}^{\mathrm{e}}_{ij}$ is the average encoding rate of the already buffered VUs in buffer i at time j obtained by

$$\bar{R}^{\mathrm{e}}_{i} = \sum_{l=1}^{\lfloor f_{ij} \rfloor} R^{\mathrm{e}}_{ij-1} + R^{\mathrm{e}}_{i}(j - \lceil f_{ij} \rceil)\left(f_{ij} - \lfloor f_{ij} \rfloor\right) \tag{4.22}$$

where f_{ij} is the number of VUs in i-th buffer at time j and N^{VU} is the number of frame per VU. In order to simplify the average encoding rate in (5.6), one may estimate it iteratively using a moving average as follows

$$\begin{aligned} \bar{R}^{\mathrm{e}}_{i1} &= R^{\mathrm{e}}_{i1} \\ \bar{R}^{\mathrm{e}}_{ij} &= \alpha R^{\mathrm{e}}_{ij} + (1-\alpha)\bar{R}^{\mathrm{e}}_{ij-1}, \end{aligned} \tag{4.23}$$

where $\alpha < 1$ is some forgetting factor.

Combining (4.20), and (5.38), one obtains

$$\tau_{ij+1} \;\; = \;\; \frac{B_{ij+1}}{\bar{R}^{e}_{ij}} = \frac{B_{ij} + \left(R^{e}_{ij} - R^{t}_{ij}\right) T}{\bar{R}^{e}_{ij}} \tag{4.24}$$

[1] MBMS service requires less than 1 second delay when switching between two video programs [oRIB08].

and from (4.19), one gets for example for $i' = 1$

$$\frac{B_{1j} + \left(R^e_{1j} - R^t_{1j}\right) T}{\bar{R}^e_{1j}} = \frac{B_{ij} + \left(R^e_{ij} - R^t_{ij}\right) T}{\bar{R}^e_{ij}}. \tag{4.25}$$

The transmission rate is then expressed by

$$R^t_{ij} = R^e_{ij} - \frac{1}{T}\left(\frac{\bar{R}^e_{ij}}{\bar{R}^e_{1j}}\left(B_{1j} + \left(R^e_{1j} - R^t_{1j}\right) T\right) - B_{ij}\right). \tag{4.26}$$

The channel rate constraint (4.18) leads to

$$\sum_{i=1}^{N} R^e_{ij} + \frac{1}{T}\sum_{i=1}^{N} B_{ij} - \frac{1}{T\bar{R}^e_{1j}}\sum_{i=1}^{N}\left(\bar{R}^e_{ij}\left(B_{1j} + \left(R^e_{1j} - R^t_{1j}\right) T\right)\right) = R^c_j,$$

$$\sum_{i=1}^{N} R^e_{ij} + \frac{1}{T}\sum_{i=1}^{N} B_{ij} - \frac{B_{1j}}{T\bar{R}^e_{1j}}\sum_{i=1}^{N}\bar{R}^e_{ij} - \frac{R^e_{1j}}{\bar{R}^e_{1j}}\sum_{i=1}^{N}\bar{R}^e_{ij} + \frac{R^t_{1j}}{\bar{R}^e_{1j}}\sum_{i=1}^{N}\bar{R}^e_{ij} = R^c_j,$$

$$R^t_{1j} = \frac{\bar{R}^e_{1j}}{\sum_{i=1}^{N}\bar{R}^e_{ij}}\left(R^c_j - \sum_{i=1}^{N} R^e_{ij} - \frac{1}{T}\sum_{i=1}^{N} B_{ij} + \left(\frac{B_{1j} + TR^e_{1j}}{T\bar{R}^e_{1j}}\right)\sum_{i=1}^{N}\bar{R}^e_{ij}\right),$$

$$R^t_{1j} = R^e_{1j} + \frac{B_{1j}}{T} + \frac{\bar{R}^e_{1j}}{\sum_{i=1}^{N}\bar{R}^e_{ij}}\left(R^c_j - \sum_{i=1}^{N} R^e_{ij} - \frac{1}{T}\sum_{i=1}^{N} B_{ij}\right).$$

A general solution of the transmission rate for $i = 1 \ldots N$ is

$$R^t_{ij} = R^e_{ij} + \frac{B_{ij}}{T} + \frac{\bar{R}^e_{ij}}{\sum_{k=1}^{N}\bar{R}^e_{kj}}\left(R^c_j - \sum_{k=1}^{N} R^e_{kj} - \frac{1}{T}\sum_{k=1}^{N} B_{kj}\right). \tag{4.27}$$

The transmission rate in (4.27) leads to a similar delays among programs. These delays are not strictly equal due to the approximation of the average encoding rate considered in (5.39). This average delay has to remain close to some reference delay τ_0, which is chosen not too large to limit the global delivery delay, but not too small to mitigate the variations with time of the R-D characteristics of video programs and of the channel rate.

At time j, the average delay deviation $\Delta\tau_j$ of each buffer delay τ_{ij} from τ_0 is

$$\Delta\tau_j = \frac{1}{N}\sum_{i=1}^{N}\left(\tau_{ij} - \tau_0\right). \tag{4.28}$$

When $\Delta\tau_j > 0$, the buffer delays are in average higher than the reference level τ_0 and the encoding rate of the next VUs should be reduced. When $\Delta\tau_j < 0$, the buffers are draining too fast and the encoding rate may be temporarily increased. As in (4.17), we evaluate an updated encoding rate constraint R_j at the buffer controller using a Proportional Integral Derivative (PID) control

$$R_j = R^c_j\left(1 - K_{\mathrm{P}}\frac{\Delta\tau_j}{T} - K_{\mathrm{I}}\sum_{k=1}^{j}\frac{\Delta\tau_k}{T} - K_{\mathrm{D}}\left(\frac{\Delta\tau_j}{T} - \frac{\Delta\tau_{j-1}}{T}\right)\right), \tag{4.29}$$

and to feed it back to the encoder controller. The stability analysis of the proposed feedback control systems is not conducted in this chapter. The study of the system stability will be provided in the next chapter for a more general case of SM control system.

3.3 Quality constraints

3.3.1 Minimum PSNR constraint

To keep an acceptable visual quality, the PSNR within a VU has to be larger than $P_{\min}$, the minimum tolerated PSNR. This leads to the constraints

$$P_{ij+k}(Q_{ij+k}^{(j)}) \geqslant P_{\min},\ i = 1 \dots N,\ k = 0 \dots W-2, \tag{4.30}$$

where P_{ij+k} is the PSNR of the i-th program at the $(j+k)$-th VU. Since the future $W-2$ VUs required to formulate the constraints for $k > 0$ have already been stored to satisfy the constraint presented in (4.5), no additional delay is introduced.

3.3.2 Smoothness constraint

Large PSNR variation between VUs may be visually annoying. The problem of providing video sequences with smooth quality variations has already been addressed in a single video encoding context in [LXLZ01, MAvdS+03], and in an SM context by [VTT07, TVT08]. Here, our aim is to provide some smoothness between successive VUs, considered as GoPs. We refer to the works of [LXLZ01, MAvdS+03] to perform smoothness within a GoP.

Our aim is to bound PSNR variations between successive VUs. This constraint may be relaxed in presence of scene changes, according to the results in [SLS08]: in case of high video activity, the bit rate (and thus the quality) may be reduced to save some bit rate for parts of the video with less activity. We assume that VUs in which a scene change occurs have been detected using the methods presented, *e.g.*, in [FL04, KW05].

At time j, the absolute value of the PSNR difference between two consecutive VUs is constrained to be less than the PSNR variation bound $\Delta P_{ij}^{\mathrm{s}}$. This bound is updated when a scene change occurs as follows

$$\Delta P_{ij}^{\mathrm{s}} = \sum_{k=-\infty}^{+\infty} S_{ij-k}^{c} h_k^{\mathrm{s}} + \Delta P_{min}^{s} \tag{4.31}$$

with

$$h_k^{\mathrm{s}} = (\Delta P_{\max}^{\mathrm{s}} - \Delta P_{\min}^{\mathrm{s}}) \exp(-\lambda |k|) \tag{4.32}$$

with $S_{ij}^{c} = 1$ if a scene change is detected in VU at time j and $S_{ij}^{c} = 0$ else. $\Delta P_{\max}^{s}$ and $\Delta P_{\min}^{s}$ are respectively the maximum and the minimum PSNR variation bounds and λ is some decay rate.

Predictive control at time j takes into account the PSNR of one past (at time $j-1$), the current, and $W-2$ future VUs. The smoothness constraint for the i-th program translates into

$$|P_{ij}(Q_{ij}^{(j)}) - P_{ij-1}(\widehat{Q}_{ij-1}^{(j-1)})| \leqslant \Delta P_{ij}^{\mathrm{s}}, \tag{4.33}$$

between time $j-1$, at which the control output $\widehat{Q}_{ij-1}^{(j-1)}$ has already been applied, and time j. Moreover, for $k = 1 \dots W-2$, the smoothness constraint becomes

$$|P_{ij+k}(Q_{ij+k}^{(j)}) - P_{ij+k-1}(Q_{ij+k-1}^{(j)})| \leqslant \Delta P_{ij+k}^{\mathrm{s}}, \tag{4.34}$$

with $i = 1 \dots N$ for the future VUs.

3.3.3 Inter-programs fairness constraint

Our aim is to provide multiplexed programs with quality levels of the same order of magnitude. For that purpose, the absolute value of the PSNR difference between two programs i and i' is constrained to be less than some PSNR discrepancy bound $\Delta P_{(i,i')j}^{\mathrm{f}}$. These bounds are such that $\Delta P_{(i,i')j}^{\mathrm{f}} = \Delta P_{(i',i)j}^{\mathrm{f}}$ for all $i, i' \in \{1 \dots N\}$. Since at scene changes, the smoothness constraint is relaxed, it is necessary to update $\Delta P_{(i,i')j}^{\mathrm{f}}$ accordingly

$$\Delta P_{(i,i')j}^{\mathrm{f}} = \sum_{k=-\infty}^{+\infty} \max\left(S_{ij-k}^{c}, S_{i'j-k}^{c}\right) h_k^{\mathrm{f}} + \Delta P_{\min}^{\mathrm{f}} \tag{4.35}$$

with

$$h_k^{\mathrm{f}} = (\Delta P_{\max}^{\mathrm{f}} - \Delta P_{\min}^{\mathrm{f}}) \exp(-\lambda |k|) \tag{4.36}$$

where $\Delta P_{\max}^{\mathrm{f}}$ and $\Delta P_{\min}^{\mathrm{f}}$ are respectively the maximum and the minimum PSNR discrepancy bounds. Then, the fairness constraint at time j translates into $N\,(N-1)\,/2$ inequality constraints

$$|P_{ij+k}(Q_{ij+k}^{(j)}) - P_{i'j+k}(Q_{i'j+k}^{(j)})| \leqslant \Delta P_{(i,i')j+k}^{\mathrm{f}}, \tag{4.37}$$

with $k = 0 \dots W-2$ and $i, i' \in \{1 \dots N\}$.

3.4 Summarized constrained optimization problem

Considering the cost function (4.3) and the constraints related to the rate (4.4) and (4.5), the minimum PSNR (4.30), the smoothness (4.33) and (4.34), and the fairness (4.37) constraints, one gets the following constrained optimization problem to solve at time j

$$\left(\widehat{\mathbf{Q}}_1^{(j)} \dots \widehat{\mathbf{Q}}_N^{(j)}\right) = \arg \max_{\mathbf{Q}_1^{(j)} \dots \mathbf{Q}_N^{(j)}} \sum_{k=j}^{j+W-2} \gamma^{k-j} \sum_{i=1}^{N} P_{ik}\left(Q_{ik}^{(j)}\right) \tag{4.38}$$

subject to

$$\begin{cases} \sum_{i=1}^{N} R_{ij}^{\mathrm{e}}(Q_{ij}^{(j)}) = R_j \\ \sum_{i=1}^{N} R_{ij+k}^{\mathrm{e}}(Q_{ij+k}^{(j)}) = E\left(R_j^{\mathrm{c}} | S_j\right) \\ P_{ij+h}(Q_{ij+h}^{(j)}) \geqslant P_{\min} \\ |P_{ij+h}(Q_{ij+h}^{(j)}) - P_{i'j+h}(Q_{i'j+h}^{(j)})| \leqslant \Delta P_{(i,i')j+h}^{\mathrm{f}}, \\ |P_{ij}(Q_{ij}^{(j)}) - P_{ij-1}(\widehat{Q}_{ij-1}^{(j-1)})| \leqslant \Delta P_{ij}^{\mathrm{s}}, \\ |P_{ij+k}(Q_{ij+k}^{(j)}) - P_{ij+k-1}(Q_{ij+k-1}^{(j)})| \leqslant \Delta P_{ij+k}^{\mathrm{s}} \\ \text{with } h = 0 \dots W-2,\ k = 1 \dots W-2 \\ \text{and } (i, i') \in \{1 \dots N\} \end{cases} \tag{4.39}$$

where $\widehat{\mathbf{Q}}_{j-1}^{j-1}$ contains the QP obtained from step $j-1$ for VU $j-1$.

The control of the transmission rate and buffer delay is described in Sections 3.2.3 and 3.2.4.

3.5 Statistical multiplexing application in an MBMS system

A typical scenario for statistical multiplexing application is the Mobile TV service delivery over evolved MBMS standard [ETS05]. Here, we briefly describe the functional architecture of the multiplexing functions. Detailed implementation issues are not addressed.

MBMS is a point-to-multipoint interface specification for 3GPP cellular networks, which is designed to provide efficient delivery of broadcast and multicast services. For broadcast transmission, a single frequency network configuration is introduced in 3GPP LTE (Long Term Evolution) specifications which enables a time-synchronization between a set of eNBs (base stations) using the same resource block.

MBMS architecture is composed of three main entities: BM-SC, MBMS-GW and MCE, see Figure 4.3. The Multicast/Broadcast Service Center (BM-SC) is a node that serves as an entry point for the content providers delivering the video sources, used for service announcements, session management.

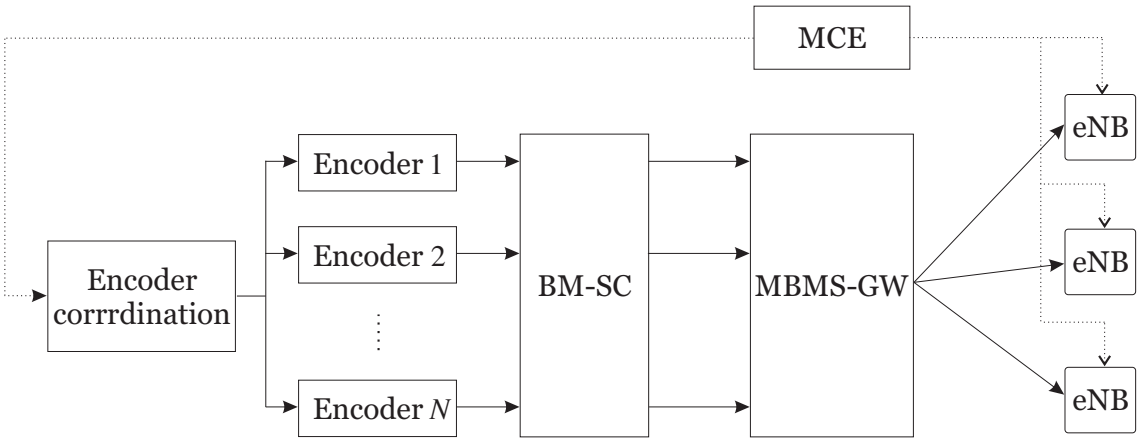


Figure 4.3: Building blocks of a processing chain for video multiplexing in MBMS

The MBMS-GW is an entity responsible for distributing the traffic across the different eNBs belonging to the same broadcast area. It ensures that the same content is sent from all the eNBs by using IP Multicast.

The Multi-cell/Multicast Coordination Entity (MCE) is a new logical entity, responsible for allocation of time and frequency resources for multi-cell MBMS transmission. As in [VH08], we assume that the MBMS-GW periodically notifies the MCE about the resource requirements of video streams so that the resources at eNBs can be re-allocated accordingly. Therefore, the BM-SC should ensure that the encoding rate of the multiplex does not violate the already allocated resources. This is obtained thanks to the proposed SM scheme.

$\Delta P^{\mathrm{s}}_{\max}$	2.5 dB	$\Delta P^{\mathrm{s}}_{\min}$	1 dB
$\Delta P^{\mathrm{f}}_{\max}$	5 dB	$\Delta P^{\mathrm{f}}_{\max}$	2 dB
$P_{\min}$	30	λ	1.25

Table 4.3: Quality constraint parameters.

4 Experimental Results

In this section, we evaluate the performance of the proposed joint encoder and buffer controller involving the solution of the constrained optimization problem in 4.38 and (4.39). Here, $N = 4$ programs are multiplexed and transmitted. Each program displays various video sequences in CIF format (Soccer (V1), Container (V2), Coast-guard (V3), and Hall (V4)) to simulate abrupt scene changes in the video program as represented in Figure 7.2. The video sequences are encoded with the H.264/AVC encoder in baseline profile at the same frame rate $F = 30$ frames/s.

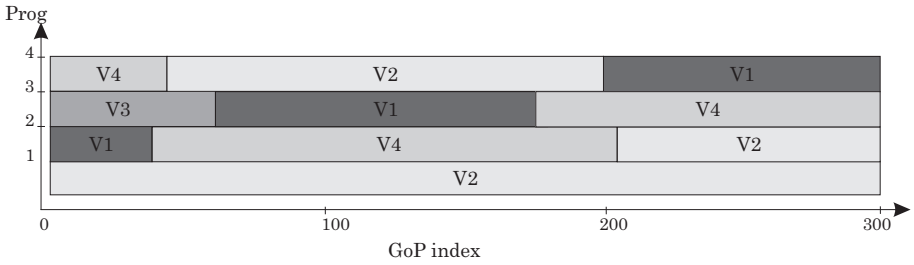


Figure 4.4: Videos transmitted over the four considered programs.

All video programs are divided into GoPs of $N_{\mathrm{f}} = 15$ frames, thus the VU (GoP) duration is $T = 0.5$ s. The first frame in the GoP is an I frame and the remaining frames are P frames. More sophisticated GoP structures may also be employed.

In the first set of experiments, we focus on the video encoder control process described in (4.38) and (4.39). The predictive control is performed using a control window of $W = 4$ GoPs and is compared to a reference scenario without predictive control ($W = 2$), but for which the smoothness constraint is still imposed. The R-D models in (5.71) and (4.1) have to be evaluated in advance for $W - 1$ GoPs with two encoding trials for each GoP. Using predictive control, an encoding delay $(W - 2)T$ is introduced. When $W = 4$, this delay is 1 s. While performing predictive control, scene changes are assumed known in advance. The quality constraints parameters values are set in Table 4.3. The considered damping ratio leads to a negligible relaxation of the PSNR bounds after two to three VUs.

The size of the buffers is taken large enough to support the large bit level variations, occurring, *e.g.*, during scene changes. Here, their size in bits is $B_{\max} = 2$ Mbits. The reference delay is taken as $\tau_0 = 1$ s, the reference buffer bit level is $B_0 = 400$ kbits. Two cases are considered: *(i)* constant channel rate as in [RBG09] and *(ii)* time-varying channel rate as in [VH08].

In what follows, the solutions of (4.38) and (4.39) involved in the control process are obtained numerically using the Matlab *fmincon* function.

4.1 Constant channel rate

In this section, we consider a constant channel rate taken as $R_j^{\mathrm{c}} = 1$ Mbit/s for all time j.

4.1.1 Buffer bit level control

We consider joint encoder and buffer control where buffers are controlled at the bit level as defined in in (4.6) and (4.7).

The parameters of the PID controller for the feedback from the buffer controller to the encoder controller are set to $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.4, 0.02, 0.25)$ for $W = 2$ and $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.5, 0.02, 0.25)$ for $W = 4$.

In fact, the parameters of the PID controller have been tuned manually using first a simplified scenario where the four multiplexed programs display only the first GoP of each video in a loop. Moreover, buffers have been assumed to be initially empty. This helps to get some steady-state after a transient behavior. The evolution of the buffer bit level for different values of the PID parameters is presented in Figure 4.5 (a) for $W = 2$ and (b) for $W = 4$.

K_{P} is tuned first to minimize the rise time to some equilibrium while having no overshoot. In Figure 4.5(b), the choice $K_{\mathrm{P}} = 0.5$ appears to be a good compromise. K_{I} is adjusted to eliminate the offset and to minimize the overshoot. Figure 4.5(b) shows that $K_{\mathrm{I}} = 0.02$ provides the best results. Finally K_{D} should provide some additional stability to the system. In Figure 4.5(b), $K_{\mathrm{D}} = 0.25$ is a reasonable choice. Figure 4.5 shows some oscillating behavior after the transient phase. This is due to the discrepancy between the model and the actual R-D characteristics and to the fact that the QPs provided by the optimization process have to be rounded before being used by the video encoders. This type of oscillatory behavior due to the quantized inputs has been considered in [Pic07] and [GHM11].

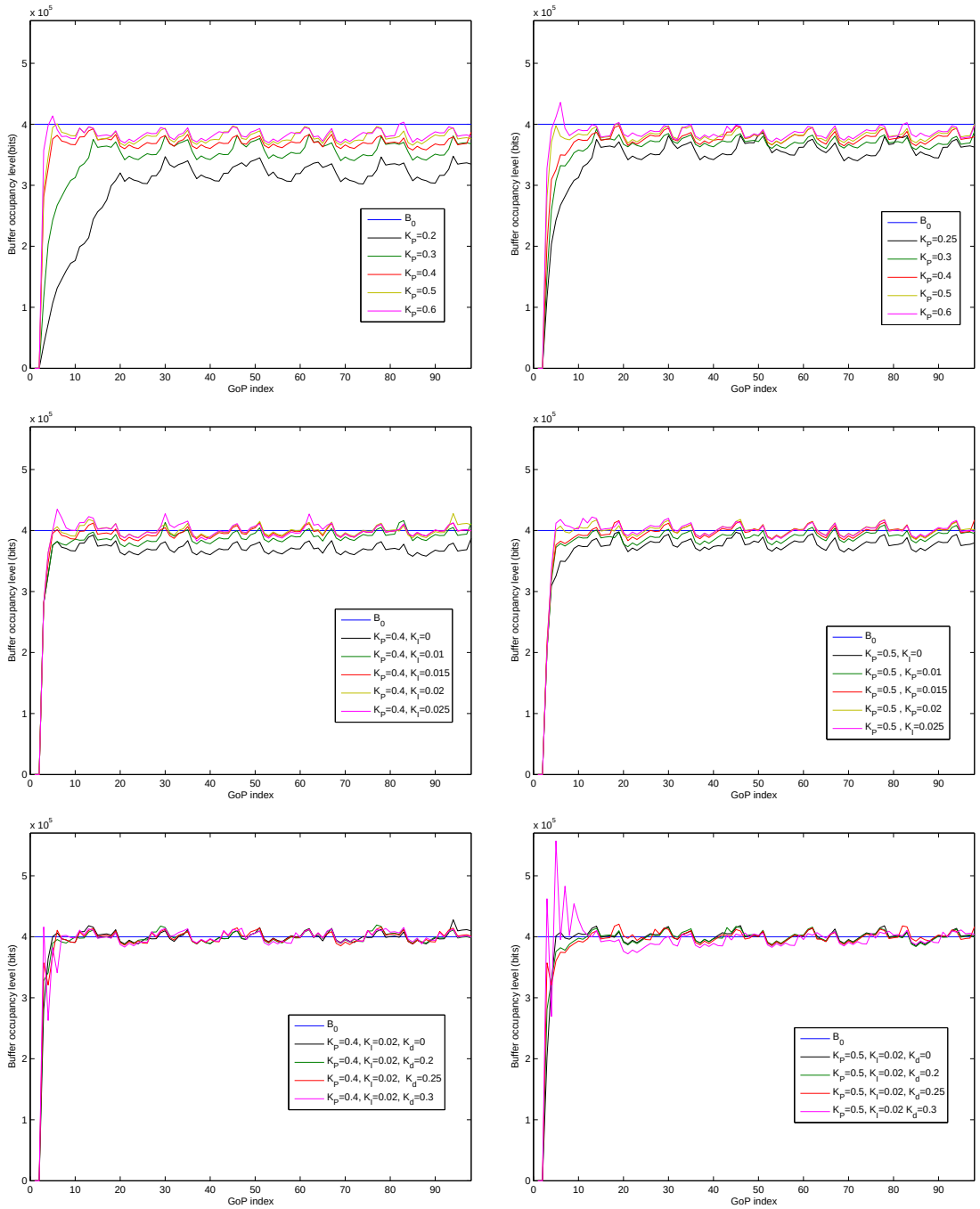
The parameters K_{P}, K_{I}, and K_{D} are then readjusted manually from those obtained using the simple case to minimize $e = (\Delta B)^2 + \sigma_B^2$ where ΔB is the average buffer bit level discrepancy and σ_B^2 is the average buffer bit level variance considering the actual programs.

4.1.1.a Rate and quality control

Figure 4.6 shows the encoding rates R_{ij}^{e} for each program, the total encoding rate $R_j^{\mathrm{e}} = \sum_{i=1}^{4} R_{ij}^{\mathrm{e}}$ and the total transmission rate $R_j^{\mathrm{t}} = \sum_{i=1}^{4} R_{ij}^{\mathrm{t}}$ for $W = 2$ (a) and $W = 4$ (b). In both cases, R_j^{t} is equal to the channel rate R_j^{c} thanks to the buffer control.

The PSNR P_{ij} of each program is represented in Figure 4.7 for $W = 2$ (a) and $W = 4$ (b). Thanks to the fairness constraint, a similar quality is obtained for all transmitted programs.

The differences between successive PSNRs $(P_{ij} - P_{ij-1})$ for each program are represented in Figure 4.8 in the following cases: without smoothness constraint, with a smoothness constraint involving only the past VU (case $W = 2$), and with a smoothness constraint involving both past and future VUs ($W = 4$). The smoothness constraints are less frequently violated when $W = 2$ (2.3 % of time) or when $W = 4$ (1.5 % of time) than without smoothness constraint (3.5 % of time). Some

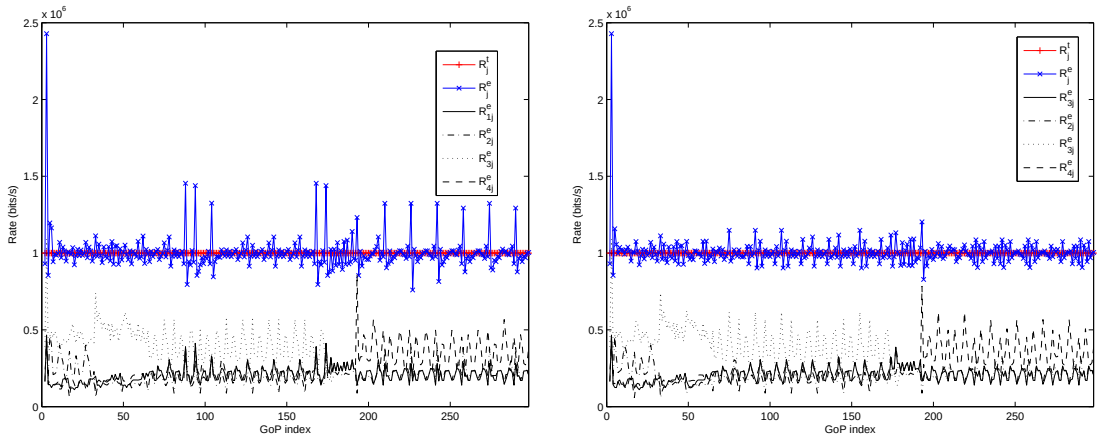


(a) (b)

Figure 4.5: Tuning of the parameters of P, PI, and PID controller with $W = 2$ (a) and $W = 4$ (b) when buffers bit levels are controlled.

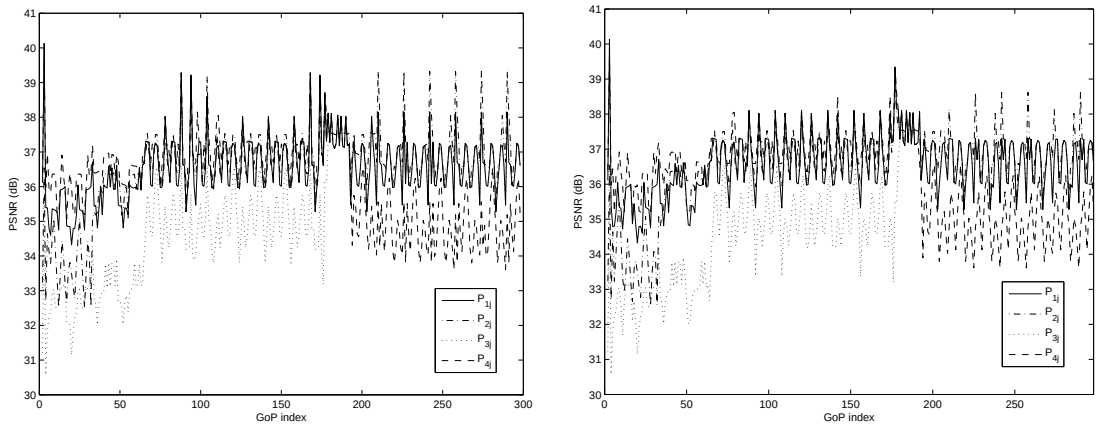
violations occur even when smoothness is requested due to the discrepancy between the R-D models used in the control process and the actual R-D characteristics of each video sequence.

Taking $W > 4$ with the same values of $\Delta P_{ij}^{\mathrm{f}}$ and $\Delta P_{ij}^{\mathrm{s}}$ does not provide any additional benefit in terms of variations of the PSNR.



(a) (b)

Figure 4.6: Evolution of the total transmission rates as well as individual and total encoding rates for the four multiplexed programs for $W = 2$ (a) and $W = 4$ (b), when buffers bit levels are controlled.



(a) (b)

Figure 4.7: Evolution of the PSNR of the four multiplexed programs for $W = 2$ (a) and $W = 4$ (b), when buffers bit levels are controlled.

4.1.1.b Performance of the buffer control

This section illustrates the performance of the buffer management. The transmission rates R_{ij}^{t} are obtained analytically as in (4.14). The actual buffer bit levels B_{ij} for each buffer are represented in Figures 4.9 and 4.10 for $W = 2$ and $W = 4$.

The corresponding buffer delay τ_{ij} obtained from (5.38) using PID parameters, $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.4, 0.02, 0.25)$ for $W = 2$ and $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.5, 0.02, 0.25)$ for $W = 4$, is represented in Figure 4.11.

Table 4.4 provides the system performance in terms of ΔB and σ_B^2 when using P, PI, and PID controllers. From Figures 4.9 and 4.10 and Table 4.4, for both cases $W = 2$ and $W = 4$, the integral term reduces the rise speed to the reference buffer bit level B_0 and reduces significantly the discrepancy between the buffer bit level and the reference level ΔB. Then, the derivative term reduces σ_B^2.

For $W = 4$, the variation of the buffer bit level is smoother than for $W = 2$. This is due to the foresighted encoding rate adaptation performed to better satisfy

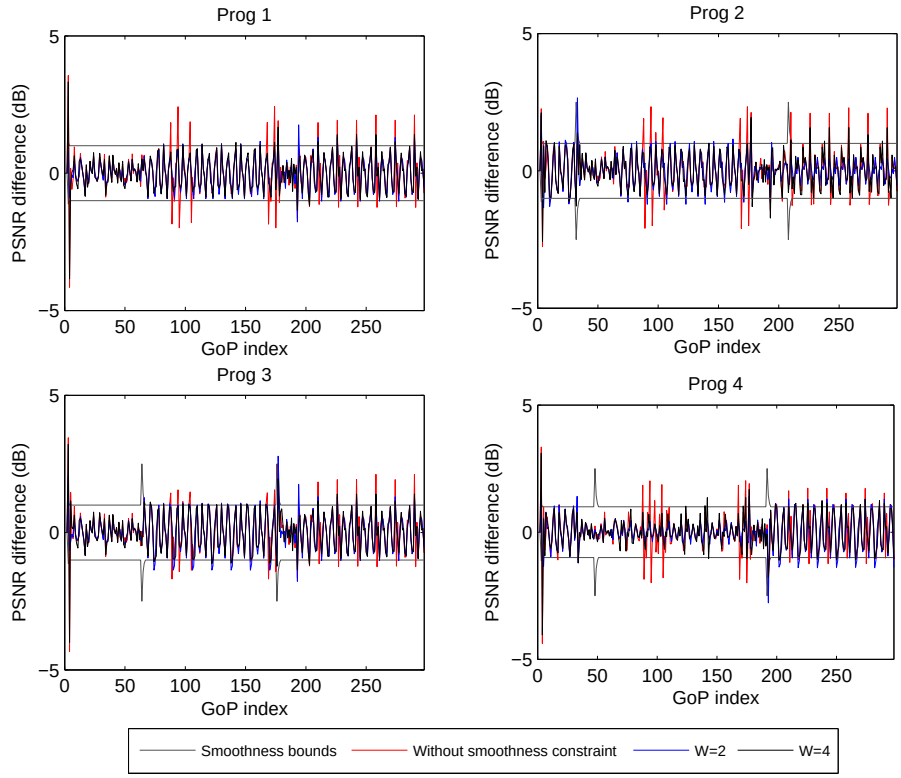


Figure 4.8: PSNR differences with smoothness constraint, for $W = 2$ and $W = 4$, when buffers bit levels are controlled.

the quality constraints.

The closed-loop control allows the buffers bit levels in the four considered buffers remaining around the reference level B_0. In addition, as demonstrated in (4.15), the four buffers have the same buffer bit levels contrarily to the buffer delay in 4.11 which is completely uncontrolled.

K_P, K_I, K_D	$W = 2$			K_p, K_I, K_D	$W = 4$		
	$\Delta B(10^4)$	$\sigma_B^2(10^9)$	$e(10^9)$		$\Delta B(10^4)$	$\sigma_B^2(10^9)$	$e(10^9)$
$0.4, 0, 0$	0.867	1.671	1.748	$0.5, 0, 0$	0.433	1.434	1.452
$0.4, 0.02, 0$	0.103	1.701	1.702	$0.5, 0.02, 0$	0.081	1.429	1.430
$0.4, 0.02, 0.25$	0.108	1.389	1.390	$0.5, 0.02, 0.25$	0.049	1.376	1.376

Table 4.4: System performance in terms of ΔB and σ_B^2 when using P, PI, and PID controllers for $W = 2$ and $W = 4$ using constant channel rate and buffer bit level control.

4.1.2 Buffer delay control

In this section, the buffers delays are controlled as described in Section 3.2.4

The parameters of the PID controller for the feedback from the buffer controller to the encoder controller are set to $(K_P, K_I, K_D) = (0.2, 0.01, 0.01)$ for $W = 2$ and to $(K_P, K_I, K_D) = (0.2, 0.01, 0.05)$ for $W = 4$.

As in the buffer bit level control, the parameters of the PID controller have been tuned manually using first a simplified scenario where the four multiplexed programs

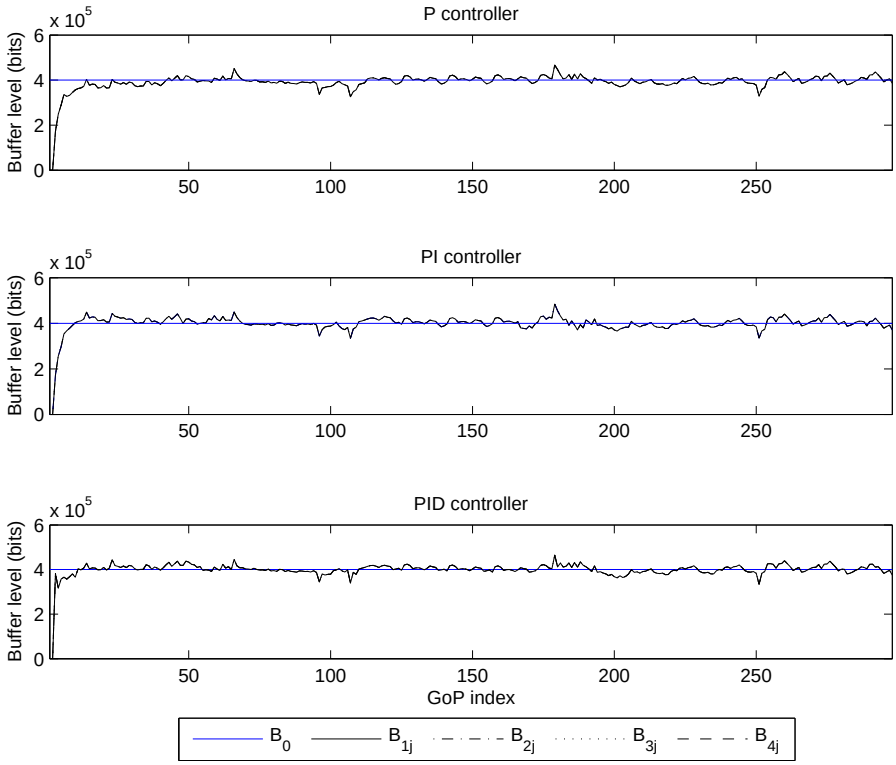


Figure 4.9: Buffer bit level evolution of the four multiplexed programs using a PID controller for $W = 2$ with constant channel rate, when buffers bit levels are controlled.

display only the first GoP of each video in a loop, where buffers have been assumed to be initially empty. The evolution of the buffer delays for different values of the PID parameters is presented in Figure 4.15 for $W = 2$ and $W = 4$.

Here, as in the buffer bit level control, K_{P}, K_{I}, and K_{D} are tuned respectively to maximize the rise speed to some equilibrium while having no overshoot and provide some additional stability to the system. In Figures 4.15(b) the choice $K_{\mathrm{P}} = 0.3$, $K_{\mathrm{I}} = 0.01$, and $K_{\mathrm{D}} = 0.2$ provides the best results for $W = 4$.

The parameters K_{P}, K_{I}, and K_{D} are then readjusted manually from those obtained using the simple case to minimize $e = (\Delta\tau)^2 + \sigma_\tau^2$ where $\Delta\tau$ is the average delay discrepancy and σ_τ^2 is the average delay variance considering the actual programs.

4.1.2.a Rate and quality control

The same rate and quality performance as that obtained when the buffers bit levels are controlled is obtained when controlling the buffer delays. The smoothness constraints are less frequently violated when $W = 2$ (2.75 % of time) or when $W = 4$ (1.8 % of time) than without smoothness constraint (4 % of time). Some violations occur even when smoothness is requested due to the discrepancy between the R-D models used in the control process and the actual R-D characteristics of each video sequence. Introducing the smoothness constraint reduces the amplitude of PSNR variations, see also Figure 4.16, where the standard deviation of the PSNR is represented for several values of the channel rate. In the three considered cases, the standard deviation decreases with the increase of the channel rate. As in the buffer

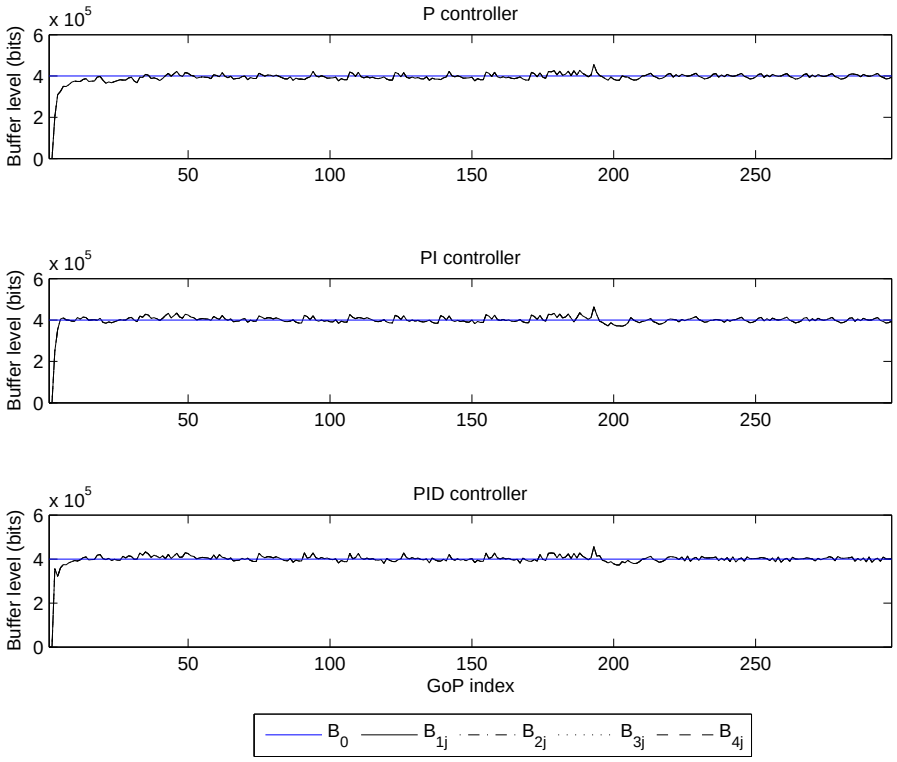
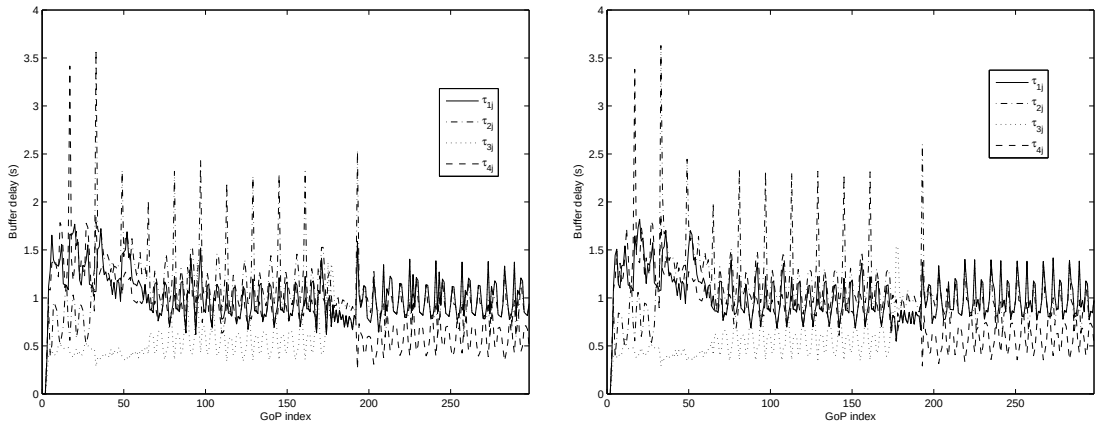


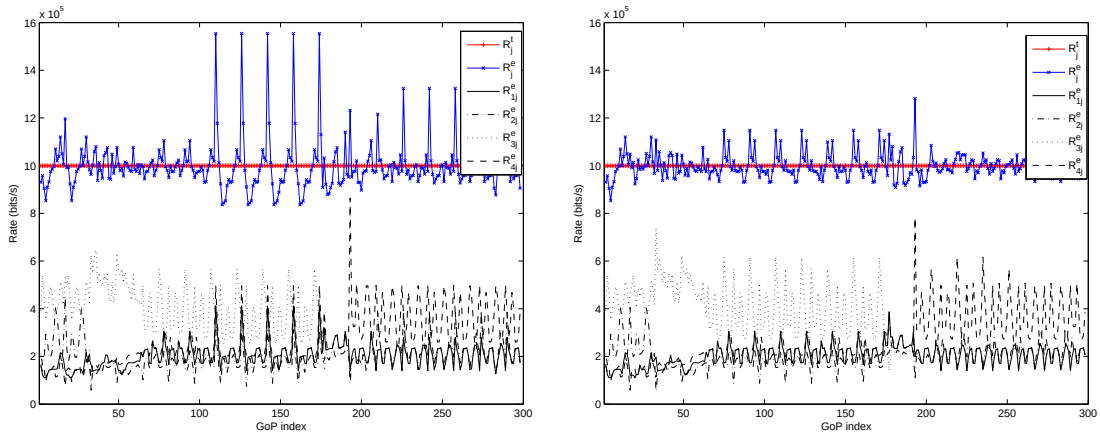
Figure 4.10: Buffer bit level evolution of the four multiplexed programs using a PID controller for $W = 4$ with constant channel rate, when buffers bit levels are controlled



(a) (b)

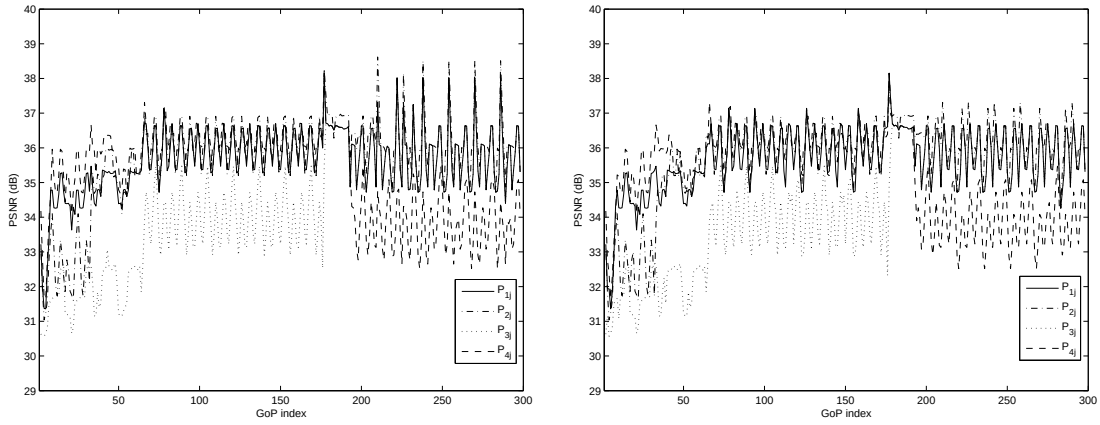
Figure 4.11: Buffer delay evolution of the four multiplexed programs using P, PI, and PID controller with $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.4, 0.02, 0.25)$ for $W = 2$ (a) and $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.5, 0.02, 0.25)$ for $W = 4$ (b) with constant channel rate, when buffers bit levels are controlled.

bit level control, taking $W > 4$ with the same values of $\Delta P_{ij}^{\mathrm{f}}$ and $\Delta P_{ij}^{\mathrm{s}}$ does not provide any additional benefit in terms of variations of the PSNR when the buffers delays are controlled.



(a) (b)

Figure 4.12: Evolution of the total transmission rates as well as individual and total encoding rates for the four multiplexed programs for $W = 2$ (a) and $W = 4$ (b), when buffers delays are controlled.



(a) (b)

Figure 4.13: Evolution of the PSNR of the four multiplexed programs for $W = 2$ (a) and $W = 4$ (b), when buffers delays are controlled.

4.1.2.b Performance of the buffer control

This section illustrates the performance of the buffer management. The transmission rates R_{ij}^{t} are obtained analytically as shown in (4.27). The actual buffer delay τ_{ij} for each buffer is represented in Figures 4.17 and 4.18 for $W = 2$ and $W = 4$.

The corresponding buffer bit level B_{ij} using PID parameters is represented in Figure 4.19 for $W = 2$ (a) and $W = 4$ (b).

Table 4.5 provides the system performance in terms of $\Delta\tau$ and σ_τ^2 when using P, PI, and PID controllers. From Figures 4.17 and 4.18 and Table 4.5, the PI control reduces significantly the delay discrepancy $\Delta\tau$. Here, the derivative term slightly reduces the delay variance σ_τ^2.

For $W = 4$, the variation of the buffer delay is smoother than for $W = 2$. This is due to the foresighted encoding rate adaptation performed to better satisfy the quality constraints. The forgetting factor is set to $\alpha = 0.7$ corresponding to the smallest average relative discrepancy (less than 2%) between the estimated delay

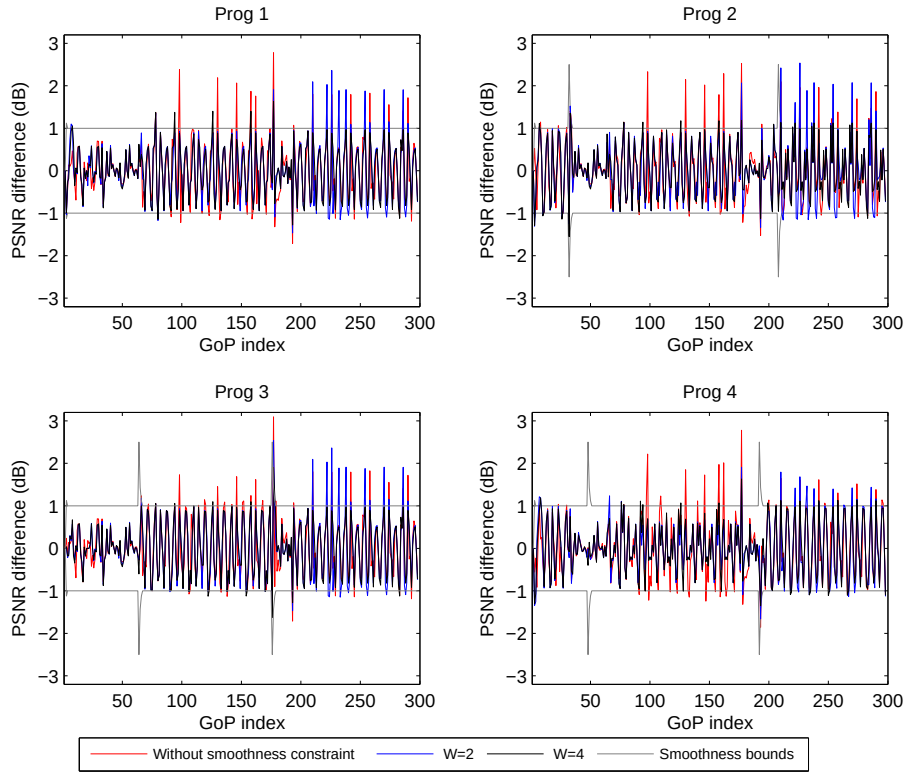


Figure 4.14: PSNR differences with smoothness constraint, for $W = 2$ and $W = 4$, when buffers delays are controlled.

K_P, K_I, K_D	$W = 2$			K_P, K_I, K_D	$W = 4$		
	$\Delta\tau$	σ_τ^2	e		$\Delta\tau$	σ_τ^2	e
$0.2, 0, 0$	0.02	0.05	0.05	$0.2, 0, 0$	0.01	0.013	0.014
$0.2, 0.01, 0$	0.01	0.04	0.04	$0.2, 0.01, 0$	0.003	0.016	0.016
$0.2, 0.01, 0.01$	0.01	0.035	0.035	$0.2, 0.01, 0.05$	0.003	0.015	0.015

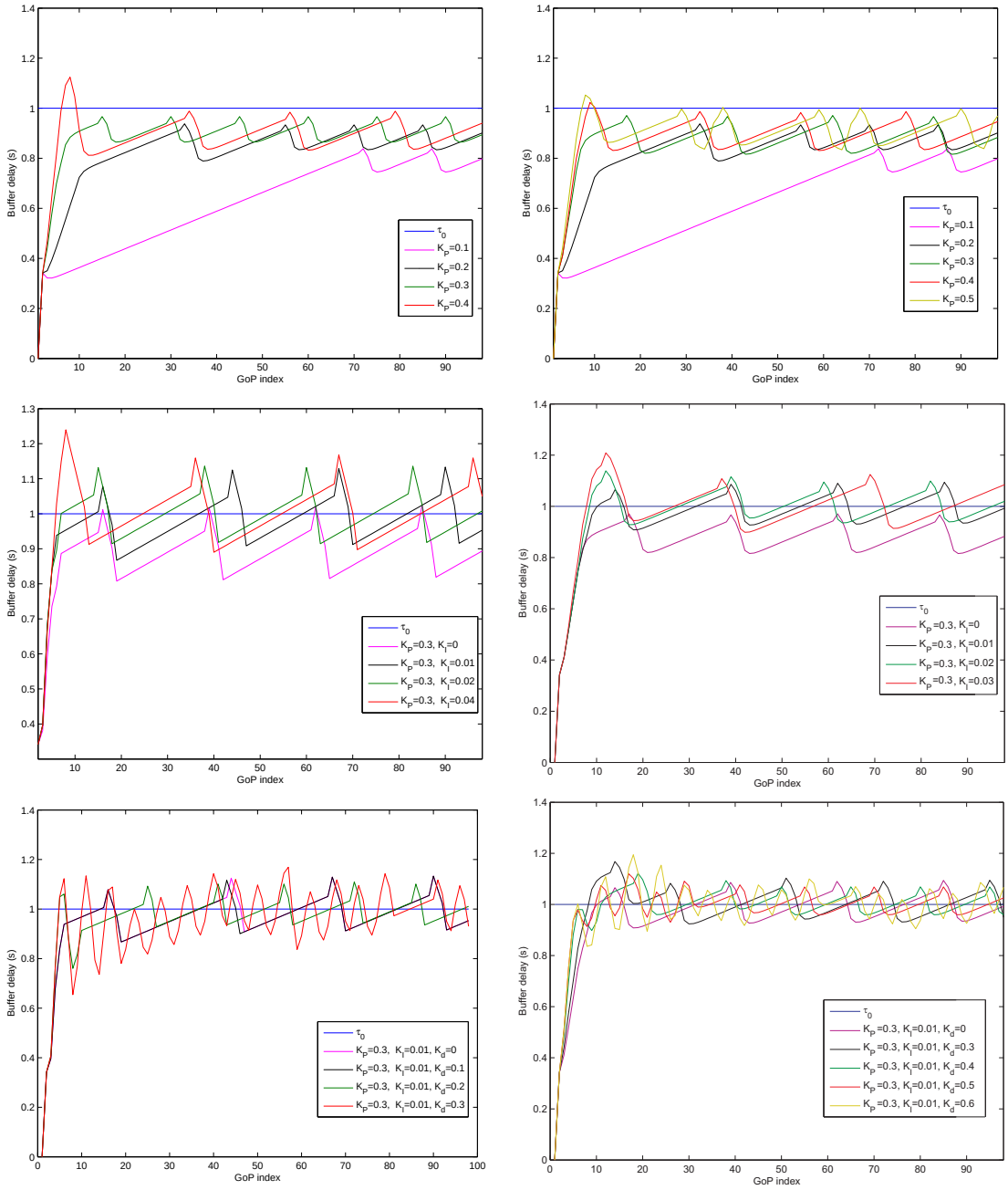
Table 4.5: System performance in terms of $\Delta\tau$ and σ_τ^2 when using P, PI, and PID controllers for $W = 2$ and $W = 4$ using constant channel rate and buffer delay control.

using (5.38) and the actual buffer delay represented in Figures 4.17 and 4.18. The closed-loop control allows the buffer delays to be of the same order of magnitude for the four multiplexed programs, even with different contents and characteristics.

In addition, although the constrained problem in (4.18) does not involve the reference delay τ_0, thanks to the PID feedback, the buffer delay in the four considered buffers remains around τ_0.

4.2 Variable channel rate

In this section, we evaluate the robustness of our proposed SM control system with respect to channel rate variations. Thus, we consider a variable broadcast channel rate. In fact, these variations may be due to concurrent services, which may leave more or less resources to the broadcast service. The rate variations are modeled as a three-state Markov chain, each state representing a rate belonging to $\mathcal{R}^c =$



(a) (b)

Figure 4.15: Tuning of the parameters of P, PI, and PID controller with $W = 2$ (a) and $W = 4$ (b), when buffers delays are controlled.

$\{800, 1000, 1200\}$ kbits/s. The channel state transition probabilities are given in the following transition matrix

$$P = \begin{bmatrix} 0.95 & 0.05 & 0 \\ 0.025 & 0.95 & 0.025 \\ 0 & 0.05 & 0.95 \end{bmatrix}. \tag{4.40}$$

One of the challenges experienced while considering predictive control and variable channel rate is that only the channel rate at time j is assumed to be known and expected rates at future time instants are evaluated as in (4.5). We evaluate, thus,

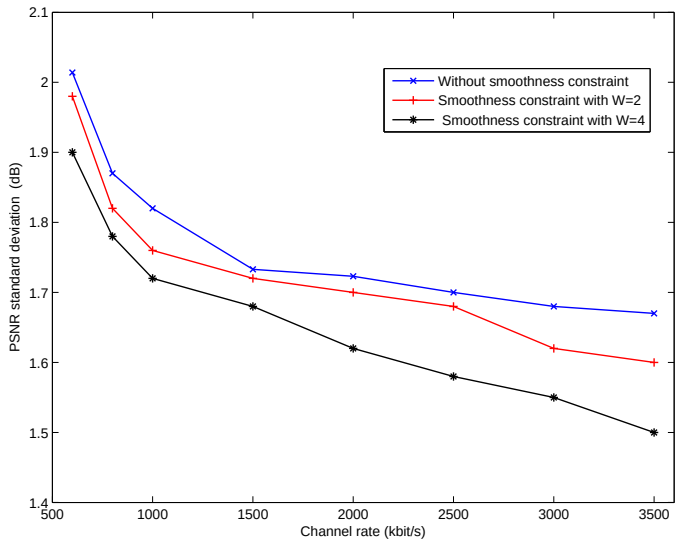


Figure 4.16: Standard deviation of the PSNR without smoothness constraint, with smoothness constraint, when $W = 2$ and $W = 4$ for different channel rates.

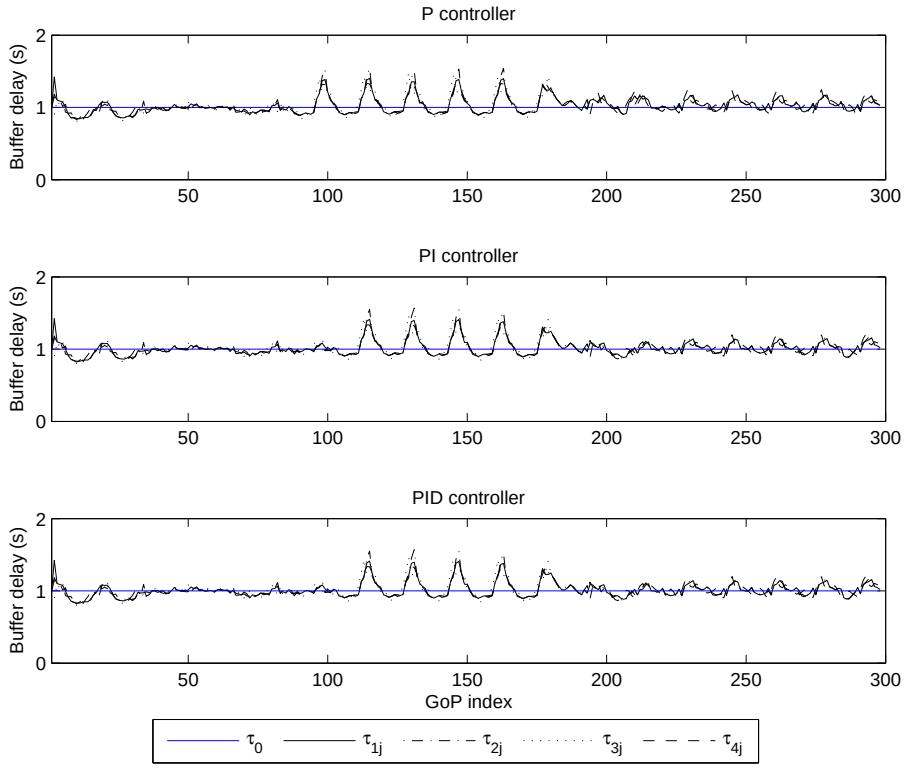


Figure 4.17: Buffer delay evolution of the four multiplexed programs using P, PI, and PID controller with $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.2, 0.01, 0.01)$ for $W = 2$ with constant channel rate, when buffers delays are controlled.

in this section the efficiency of the control system while using estimated channel rate values. Here also we consider both buffer bit level and buffer delay control cases.

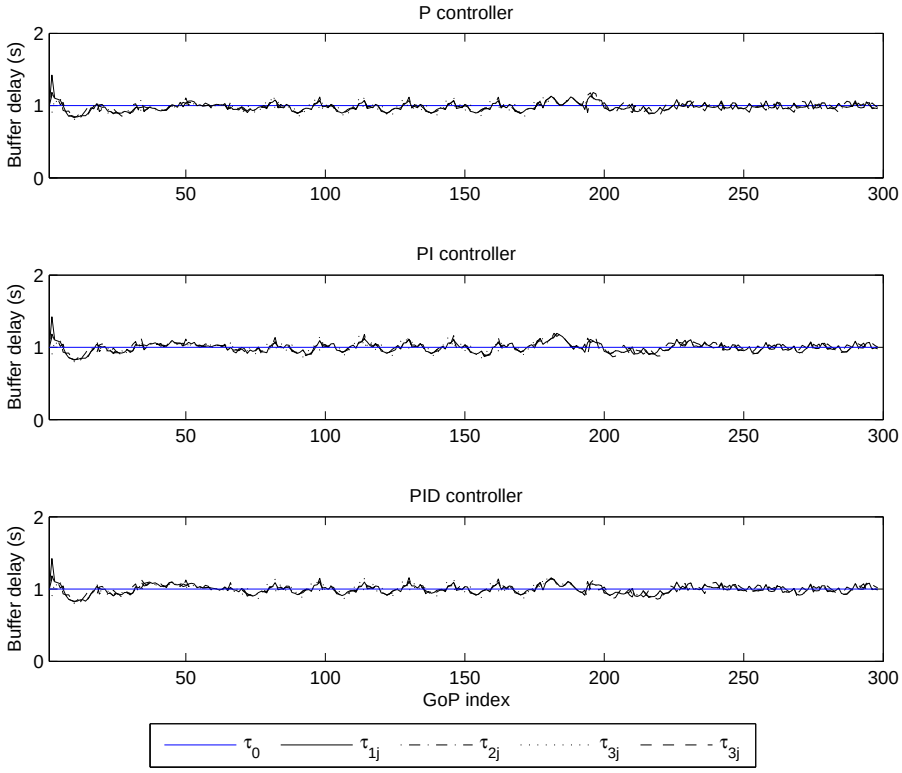
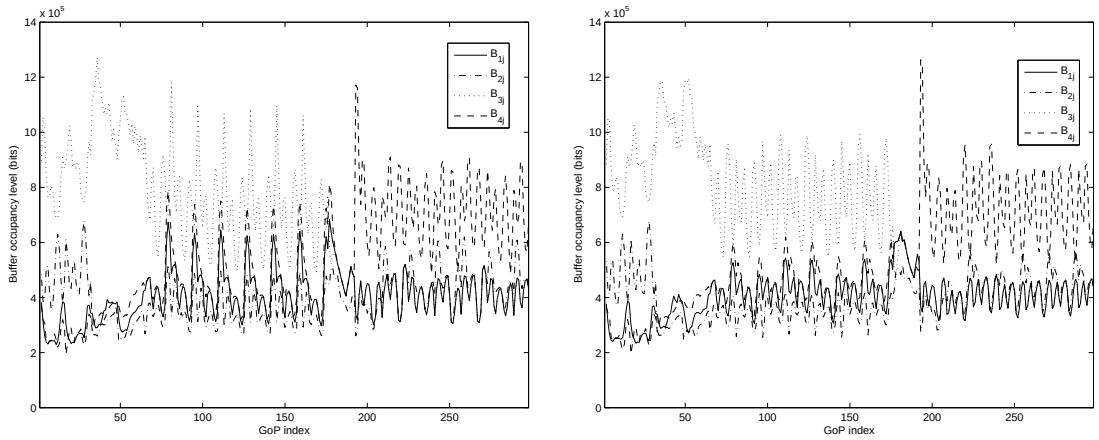


Figure 4.18: Buffer delay evolution of the four multiplexed programs using P, PI, and PID controller with $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.2, 0.01, 0.05)$ for $W = 4$ with constant channel rate, when buffers delays are controlled.



(a) (b)

Figure 4.19: Buffer bit level evolution of the four multiplexed programs using PID controller with $W = 2$ (a) and $W = 4$ (b) in the encoder control process with constant channel rate, when buffers delays are controlled.

4.2.1 Buffer bit level control

The same values for the PID parameters as in Section 4.1.1 have been used.

Figure 4.20 shows the encoding rates R^{e}_{ij} for each program, the total encoding rate $R^{\mathrm{e}}_{j} = \sum_{i=1}^{4} R^{\mathrm{e}}_{ij}$, and the total transmission rate $R^{\mathrm{t}}_{j} = \sum_{i=1}^{4} R^{\mathrm{t}}_{ij}$ for $W = 2$ (a)

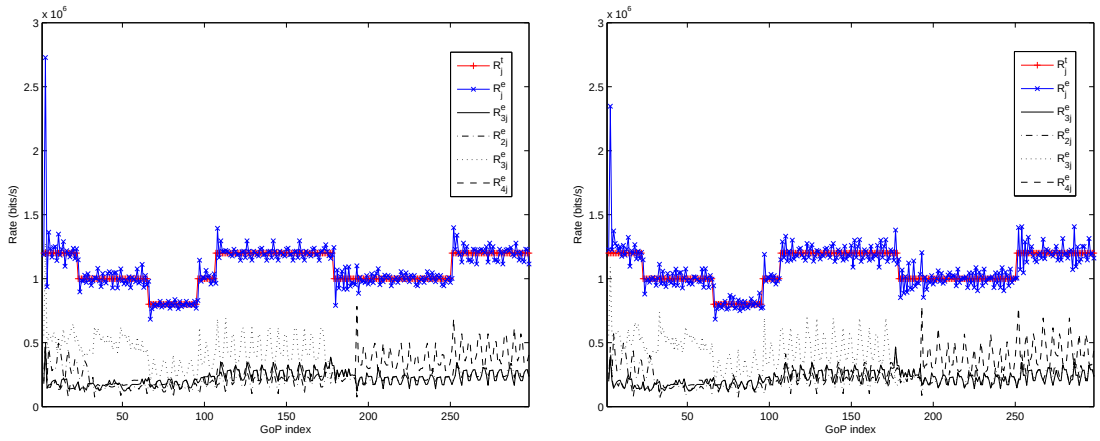


Figure 4.20: Evolution of the total transmission rates as well as individual and total encoding rates for the four multiplexed programs when $W = 2$ (a) and $W = 4$ (b) with variable channel rate, when buffers bit levels are controlled.

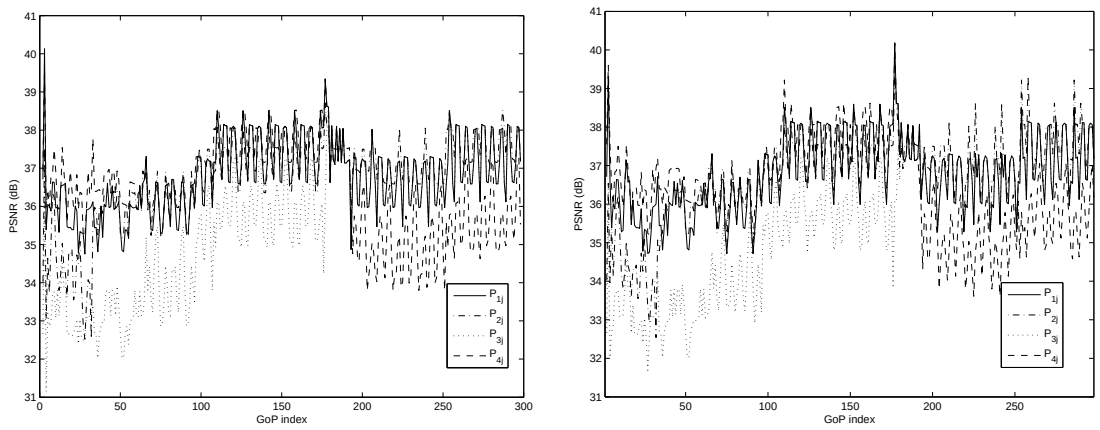


Figure 4.21: Evolution of the PSNR of the four multiplexed programs when $W = 2$ (a) and $W = 4$ (b) with variable channel rate, when buffers bit levels are controlled.

and $W = 4$ (b). The total transmission rate is equal to the channel rate. When the channel rate varies, the system is able to adapt the encoding parameters so that the total encoding rate satisfy the updated rate constraint and the smoothness and fairness constraints are satisfied. We notice that our proposed control system is robust to variations of the characteristics of the video contents and of the channel rate.

The PSNR variations resulting from the encoder control process are represented in Figure 4.21 for the four multiplexed programs for $W = 2$ (a) and $W = 4$ (b).

The performance in terms of ΔB, σ_B^2, and e for the buffer control are provided in Table 4.6, showing a good robustness of the PID parameters with respect to variations of the channel rate. The bit level in each buffer is represented in Figure 4.22 for $W = 2$ in Figure 4.23 for $W = 4$ using P, PI, and PID controllers. Similarly to the constant channel rate case, for $W = 4$, the variation of the buffer bit level is smoother than for $W = 2$.

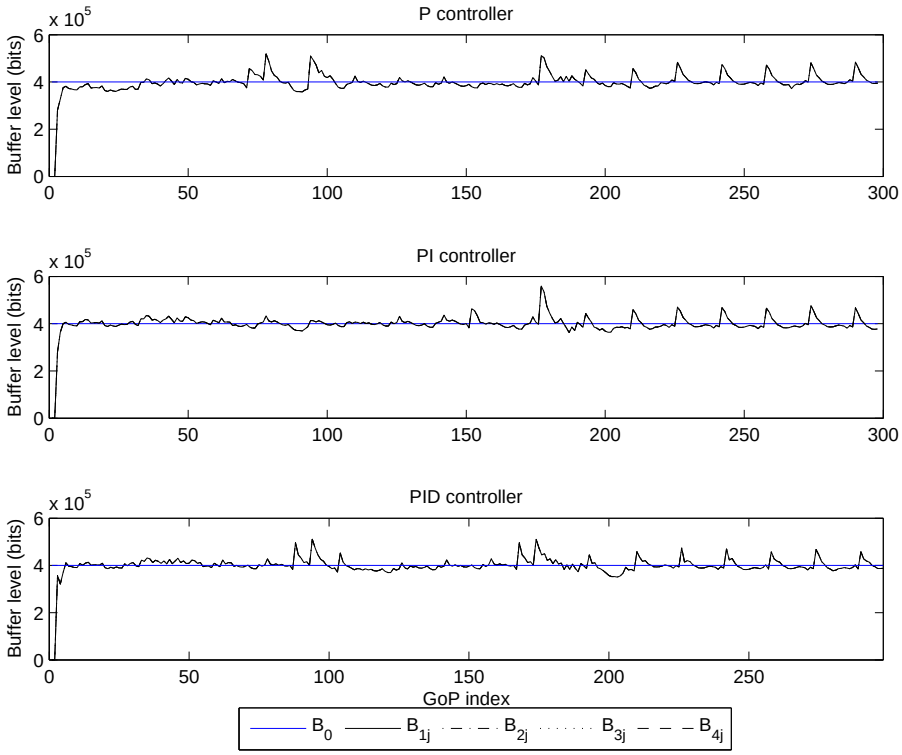


Figure 4.22: Buffer bit level evolution of the four multiplexed programs using P, PI, and PID controller with $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.4, 0.02, 0.25)$ for $W = 2$ with variable channel rate, when buffer bit levels are controlled.

$K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}$	$W = 2$			$K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}$	$W = 4$		
	$\Delta B(10^4)$	$\sigma_B^2(10^9)$	$e(10^9)$		$\Delta B(10^4)$	$\sigma_B^2(10^9)$	$e(10^9)$
$0.4, 0, 0$	3.854	2.505	3.987	$0.5, 0, 0$	3.313	1.898	2.993
$0.4, 0.02, 0$	0.978	2.925	3.019	$0.5, 0.02, 0$	1.002	2.214	2.314
$0.4, 0.02, 0.25$	0.972	2.559	2.653	$0.5, 0.02, 0.25$	1.007	1.972	2.072

Table 4.6: System performance in terms of ΔB and σ_B^2 when using P, PI, and PID controllers for $W = 2$ and $W = 4$ using variable channel rate and buffer bit level control.

4.2.2 Buffer delay control

Figure 4.24 shows the encoding rates R_{ij}^{e} for each program, the total encoding rate $R_j^{\mathrm{e}} = \sum_{i=1}^{4} R_{ij}^{\mathrm{e}}$, and the total transmission rate $R_j^{\mathrm{t}} = \sum_{i=1}^{4} R_{ij}^{\mathrm{t}}$ for $W = 2$ (a) and $W = 4$ (b). In both cases $W = 2$ and $W = 4$, the encoding rate is updated according to (4.29) to allow an efficient use of the available channel rate. The same values for the PID parameters as in Section 4.1.2 have been used.

The total transmission rate equals the channel rate. When the channel rate varies, the system is able to adapt the encoding parameters so that the total encoding rate satisfy the updated rate constraint and the smoothness and fairness constraints are satisfied. The evolution of the PSNR when buffers bit levels are controlled is represented in Figure 4.25 for the four multiplexed programs for $W = 2$ (a) and $W = 4$ (b). Similarly to the constant channel rate case, the predictive control allows reducing the PSNR standard deviation from 2.15 dB without predictive control to

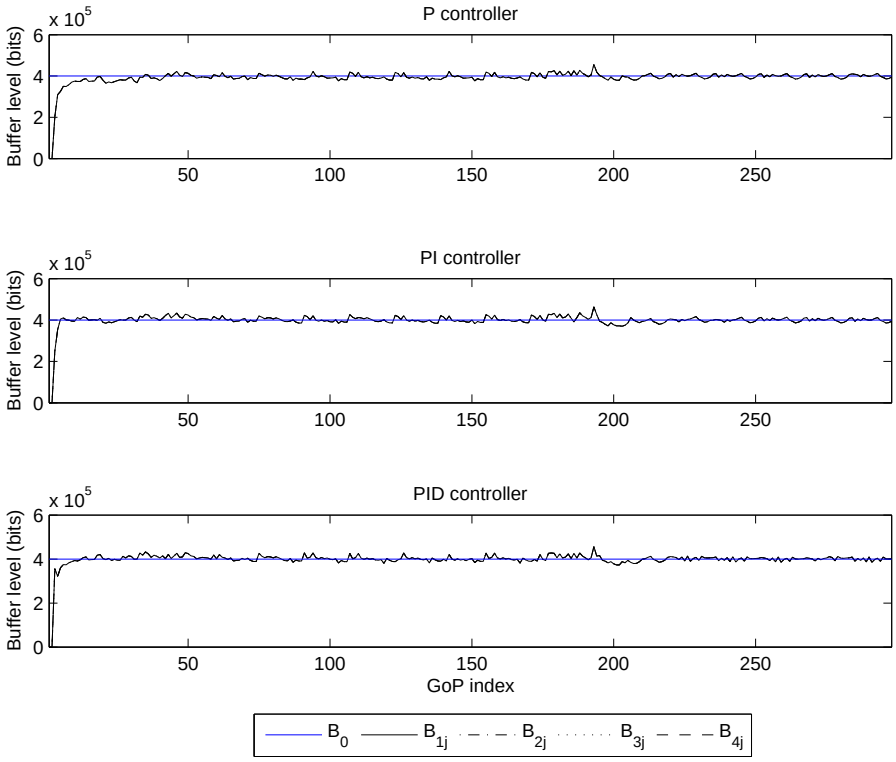
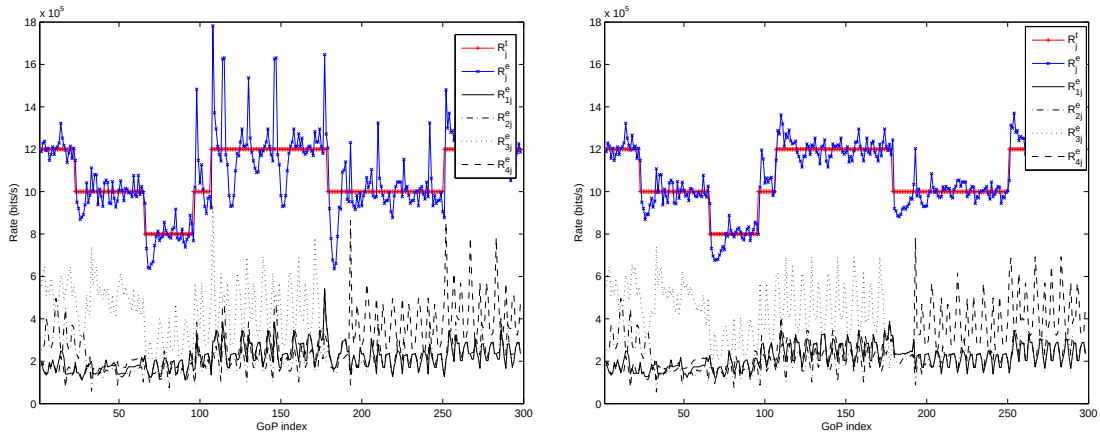


Figure 4.23: Buffer bit level evolution of the four multiplexed programs using P, PI, and PID controller with $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.5, 0.02, 0.25)$ for $W = 4$ with variable channel rate, when buffer bit levels are controlled.

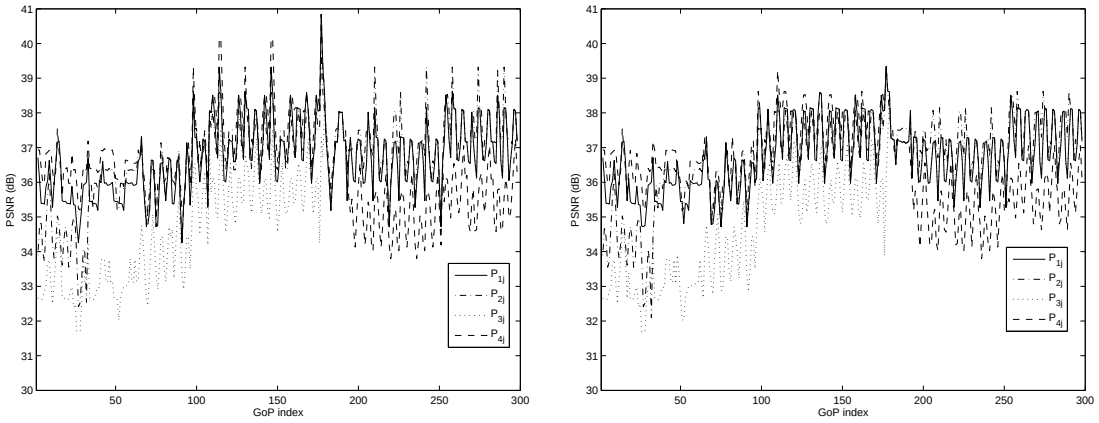


(a) (b)

Figure 4.24: Evolution of the total transmission rates as well as individual and total encoding rates for the four multiplexed programs when $W = 2$ (a) and $W = 4$ (b) with variable channel rate, when buffers delays are controlled.

1.8 dB.

The performance in terms of $\Delta\tau$, σ_τ^2, and e for the buffer control are provided in Table 4.7, showing a good robustness of the PID parameters with respect to variations of the channel rate. The buffer delays in each buffer are represented in Figure 4.26 for $W = 2$ and in Figure 4.27 for $W = 4$ using P, PI, and PID controllers. As in the previous cases, the integral and the derivative terms allows reducing the



(a) (b)

Figure 4.25: Evolution of the PSNR of the four multiplexed programs when $W = 2$ (a) and $W = 4$ (b) with variable channel rate when buffers delays are controlled.

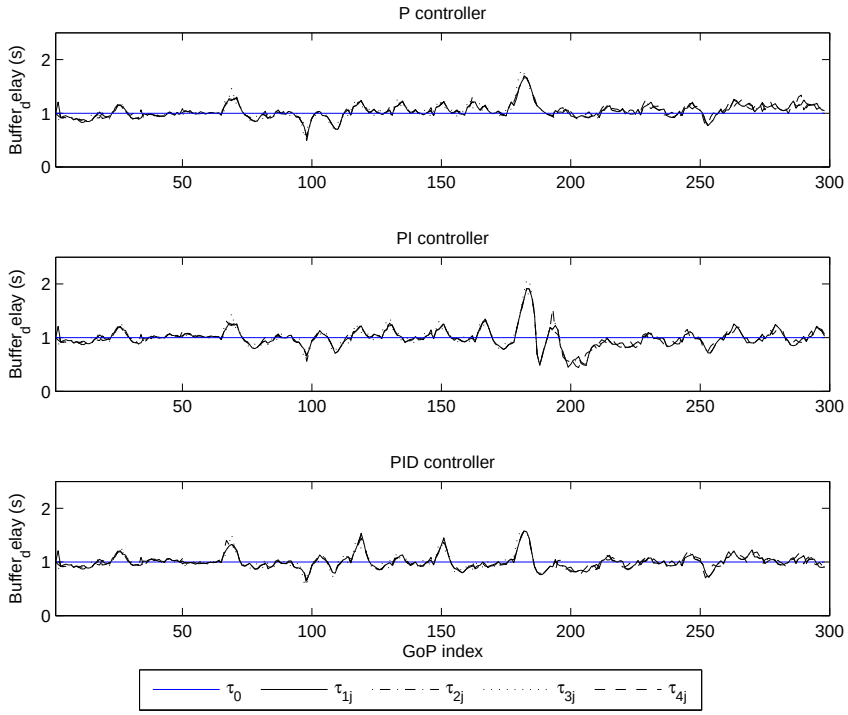


Figure 4.26: Buffer delay evolution of the four multiplexed programs using P, PI, and PID controller with $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.2, 0.01, 0.01)$ for $W = 2$ with variable channel rate, when buffers delays are controlled.

buffer delay deviation and variations.

5 Conclusion and open issues

A predictive controller for an SM system using H.264/AVC video encoders has been presented in the context of video broadcasting. The proposed system performs a closed-loop regulation of the encoders and the buffers using a PID feedback. Control

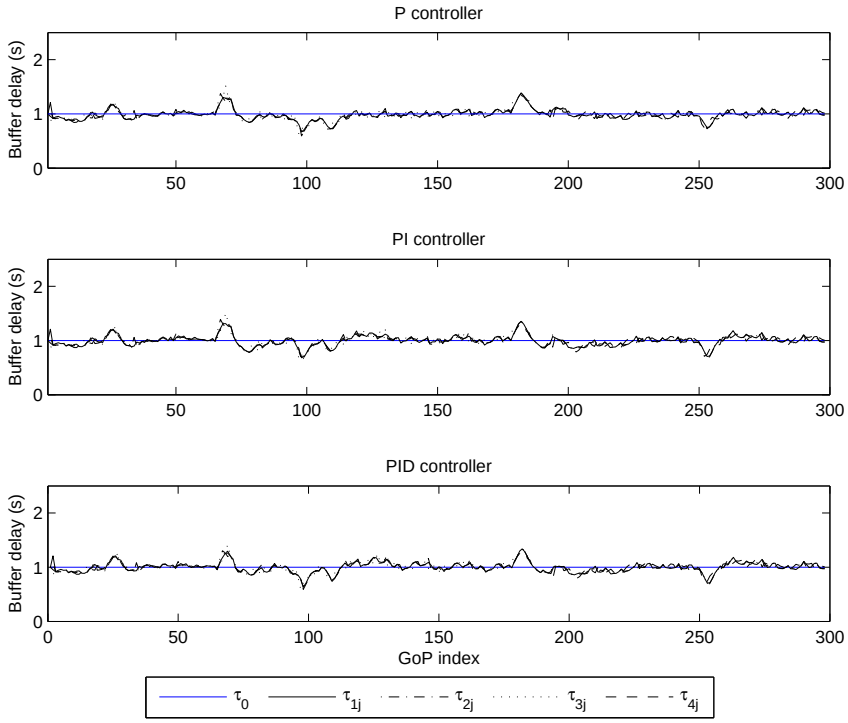


Figure 4.27: Buffer delay evolution of the four multiplexed programs using P, PI, and PID controller with $(K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}) = (0.2, 0.01, 0.05)$ for $W = 4$ with variable channel rate, when buffers delays are controlled.

$K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}$	$W = 2$			$K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}$	$W = 4$		
	$\Delta\tau$	σ_τ^2	e		$\Delta\tau$	σ_τ^2	e
$0.2, 0, 0$	0.03	0.08	0.08	$0.2, 0, 0$	0.01	0.03	0.03
$0.2, 0.01, 0$	0.001	0.13	0.13	$0.2, 0.01, 0$	0.001	0.04	0.04
$0.2, 0.01, 0.01$	0.001	0.06	0.06	$0.2, 0.01, 0.05$	0.001	0.03	0.03

Table 4.7: System performance in terms of $\Delta\tau$ and σ_τ^2 when using P, PI, and PID controllers for $W = 2$ and $W = 4$ using variable channel rate and buffer delay control.

accounts for the channel rate variations by distributing the available channel rate among the encoders while satisfying minimum quality, smoothness, and fairness constraints. A similar and small buffer delay for all multiplexed programs is also targeted.

The performance of the proposed system has been evaluated via simulations at GoP level and compared with a reference control scheme where only regulation with respect to the past GoP is performed. Experimental results with constant and variable channel rate show that thanks to the predictive and to the closed-loop control of the encoders and of the buffers, the channel is efficiently used, the video quality constraints are satisfied as well as the constraints on the buffer delays. Moreover, predictive control decreases the intra-program quality variations compared to the non-predictive control.

The adaptation of the proposed SM at the frame level, as in [HW08], requires dependent R-D models, such as those described in [LO98], [MVHB04b], or the IRS-

AR model proposed in Chapter 3. Such models $R_{ij}(Q_{ij}, Q_{ij-1})$ and $D_{ij}(Q_{ij}, Q_{ij-1})$ take into account the impact of the chosen QP in the reference frame on the rate and the distortion of its corresponding predicted frame. The price to be paid is a much increased modeling complexity than with a GoP-level control.

In addition, the proposed SM is a centralized system. In such system, to satisfy video quality fairness among programs, it is required to share continuously video quality information between the multiplexed programs. Video servers can be centralized in a service platform or they can be located in separate places. In the last case, a distributed rate control scheme is desirable. In Chapter 5, we propose a decentralized control technique able to satisfy some video quality fairness constraint among programs and requiring no exchange between servers. The stability analysis of the SM system is also addressed in Chapter 5.

Chapter 5

Distributed Statistical Multiplexing

1 Introduction

In Chapter 4, a statistical multiplexing (SM) system is proposed. The control is performed in a centralized way, where encoders or providers of the multiplexed programs should share continuously video quality information in order to satisfy the quality fairness constraint.

In this chapter, we propose a partly distributed SM system able to satisfy some video quality fairness constraint among programs without requiring exchange of information between servers. The bandwidth allocation among programs is centralized and done within some Media Aware Network Element (MANE) taking into account a quality fairness constraint. The MANE feeds back to each video server the level of its associated buffer. Thus, each video server adjusts its encoding rate independently from the others by adapting its rate-distortion trade-off so that its buffer reaches some reference level in bits or in seconds. The SM problem is modeled as a feedback control system for the bandwidth and for the encoding rate. Guidelines are provided for choosing parameters for the proposed controllers and the stability region of the system is characterized. Experimental results show that using synthetic Gaussian and video sources, an equilibrium is reached and similar video quality among the multiplexed programs is obtained. Some oscillating behavior is noticed using video sources due to the discrepancy between the model and the actual R-D characteristics. Robustness of the proposed system as well as the oscillatory behavior due to the quantized inputs will be considered in future works. Works conducted in this chapter has led to the submission of a patent application [CSK11a]. This work were also published in the proceeding of Gretsi [CSK11b] and submitted to the IEEE ICC 2012 [CSK12].

1.1 Related works

The rate-distortion (R-D) control of the SM system proposed in Chapter 4, is performed by adjusting dynamically the encoding parameters of the source coders. This may also be performed by transcoding the encoded video to match the R-D target fixed by the control scheme as in [RBG08]. With H.264/SVC, the encoded streams are organized into quality layers and the number of transmitted layers for each frame may be adjusted [JBT+08, SVHR10]. In both cases, to satisfy video quality fairness

among programs, it is required to share continuously video quality information between the multiplexed programs and thus between different servers. In a closed loop system each encoder reports the complexity of the video to the central controller which provides to each encoder an appropriate bit rate to use. This technique enables the encoders to adjust their bit rate to the complexity of the video and to the available transmission rate.

To offer a live Internet TV on a single carrier jointly multiplexed with the classical TV channels, a distributed video rate control solution is required to support the fact that the different video sources are located in separate places.

A control-theoretic approach has been considered, *e.g.*, in [HMM09] and [HM09] to address the problem of rate control for video streaming. The end-to-end multi-video sessions steaming problem is modeled as a feedback control system. A Proportional (P) controller is developed in [HMM09] to stabilize the received video quality as well as the bottleneck link queue for both homogeneous and heterogeneous video systems. In [HM09], a proportional-integral (PI) controller is considered. Robustness and stability of the PI rate control are proved. In [HMM09] and [HM09], the rate control is performed in a centralized way where the rate and the distortion characteristics of the considered video sessions are exploited to estimate the accurate encoding rate for the next frame of each session.

The problem of remotely implemented control law is arisen in [WCdWGA07] leading to the problem of stabilizing an open-loop unstable system with time-varying delay. The problem of remote stabilization via communication networks is considered with an explicit use of the average network dynamics and an estimation of the average delay dynamics in the control law. The control law does not address video transmission issue, so no quality constraint on the transmission data is considered.

Here, we propose a partially distributed control system to address the challenge of separate locations of heterogeneous video servers with no information exchange between them. The encoding rate control is performed individually at each encoder. Only the bandwidth allocation control is performed in a centralized way. In addition, accounting for the quality fairness constraint makes the stability study of the system more difficult since the control chain for all programs becomes coupled.

In [Mot09], an SM system is developed to control the overall bandwidth allocation and improve the compression efficiency. Collocated as well as spatially spread servers are supported. Control parameters and encoded streams are delivered over the network infrastructure enabling remote encoders to participate in the SM system. In [Har11], an SM system which connects encoders and multiplexers via a switched IP network allows collocated and specially spread servers to be efficiently part of the multiplexing system. Nevertheless, SM systems in [Mot09] and [Har11] allow distributed video encoders to be multiplexed but the rate control process is centralized. In addition, no quality constraint is considered among the required video encoding constraints.

1.2 Main contributions

We consider a MANE in a network element, *e.g.*, proxy of the base station, fed by several remote media servers. The MANE performs a centralized bandwidth allocation among programs. It takes into account quality information of the encoded

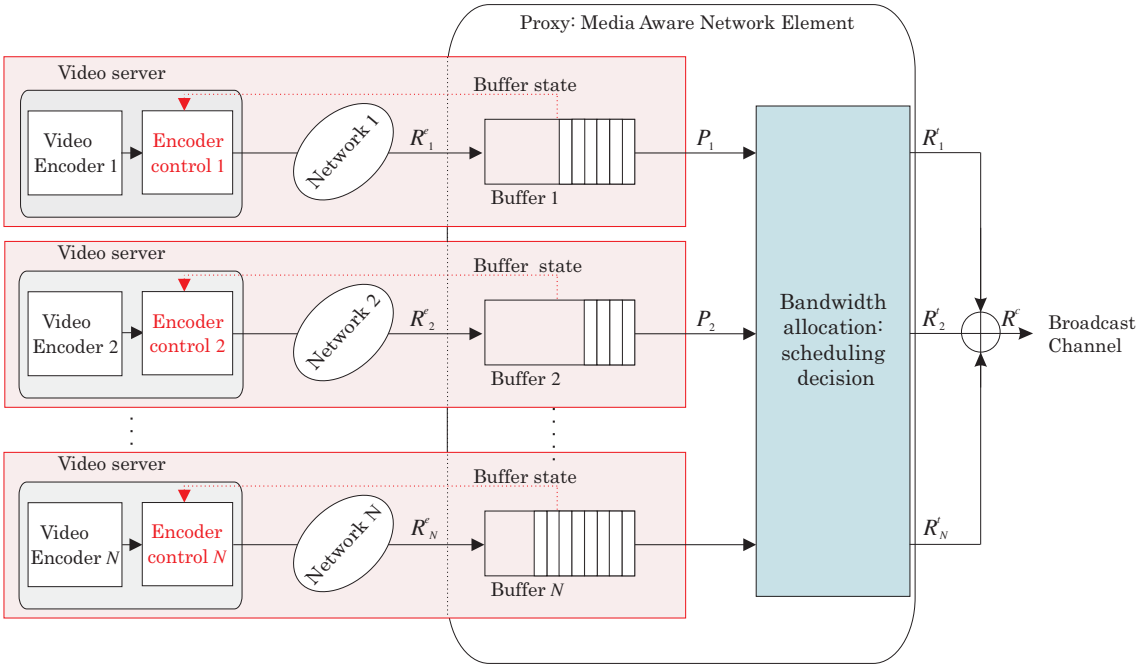


Figure 5.1: Statistical multiplexing system of video streams provided by remote servers.

videos to try to satisfy some fairness constraint. Each video server is controlled independently from the others, requiring no exchange between servers. Encoding rate control is performed via a feedback control system. We adopt a control-theoretic approach and the stability of this system is analyzed using discrete-time state space equations.

2 Problem statement

We consider a typical broadcast system in which N video programs are encoded and transmitted in parallel over a communication channel allowing a transmission rate R^{c}, see Figure 5.1. $V_i(j)$ is the j-th Video Unit (VU) in the i-th video program. A VU may be a group of samples of a Gaussian source, or a Group of Pictures (GoP). Control at the frame level is considered in Section 5. All VUs are assumed having the same duration T. The frame rate F is assumed constant with time and identical for all multiplexed programs. At time index j, the j-th encoded VUs are fed by all video encoders to the MANE. For each video stream $i \in \{1 \dots N\}$ an individual encoder rate control process is performed per VU leading to a target encoding rate $R_i^{\mathrm{e}}(j)$ that has to be such that all buffers in the MANE have the same level in bits or in seconds and a Peak Signal-to-Noise Ratio (PSNR) $P_i(j)$ for each video $i \in \{1 \dots N\}$ at time j at the same order of magnitude. Each encoded VU is then transmitted over a wire-line network and stored in its corresponding buffer of the MANE. Here, we consider no post encoder buffer, so each encoded VU is assumed immediately available at the MANE to the bandwidth allocation control process. The available transmission rate R^{c} should be distributed among the N programs, in such a way that all programs have a comparable quality. We assume that the PSNR

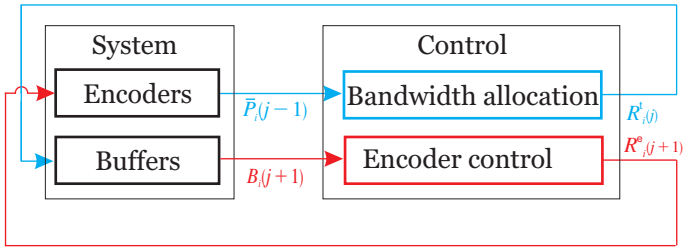


Figure 5.2: Closed-loop of our proposed control system.

information associated to the encoded VU is also transmitted and is available at the bandwidth allocation control process. Communication delays between the MANE and the encoders are not considered in this work and should be considered in future work.

3 Proposed solution

In this section, we propose a semi-distributed solution for the proposed problem. The idea is to regulate the video flow using two controllers. The first controller performs distributed *encoder control* individually for each program using buffer level information in bits or in seconds depending on the target buffer constraint. The second controller performs a centralized *bandwidth allocation*targeting a comparable video quality among the N programs is targeted using PSNR information of the last encoded and stored VU in the buffers.

The proposed control system is in charge of determining the encoding rate to encode the next VU of each video programs as well as the transmission rate at which the buffer will be drained as shown in Figure 5.2. The control process blocks in Figure 5.2 consists of two controllers, detailed in Figure 5.3.

Buffer levels information are immediately fed back to each video server that uses it to adapt its R-D trade-off so that the buffer within the MANE reaches some reference level in bits or in seconds. These two control processes are detailed in what follows.

3.1 Centralized bandwidth allocation

We consider N video programs encoded and transmitted in parallel over a communication channel. N buffers associated to each programs are located in the MANE. The evolution of the level in bits of the buffer (in the MANE) of the i-th program between time j and time $j+1$ is

$$B_i(j+1) = B_i(j) + \left(R_i^{\mathrm{e}}(j) - R_i^{\mathrm{t}}(j)\right) T. \tag{5.1}$$

where $R_i^{\mathrm{e}}(j)$ and $R_i^{\mathrm{t}}(j)$ are respectively the encoding rate and the allocated rate of program i evaluated at time j.

At each time j, the available transmission rate R^{c} is distributed among video programs in a centralized way to meet the quality fairness constraint.

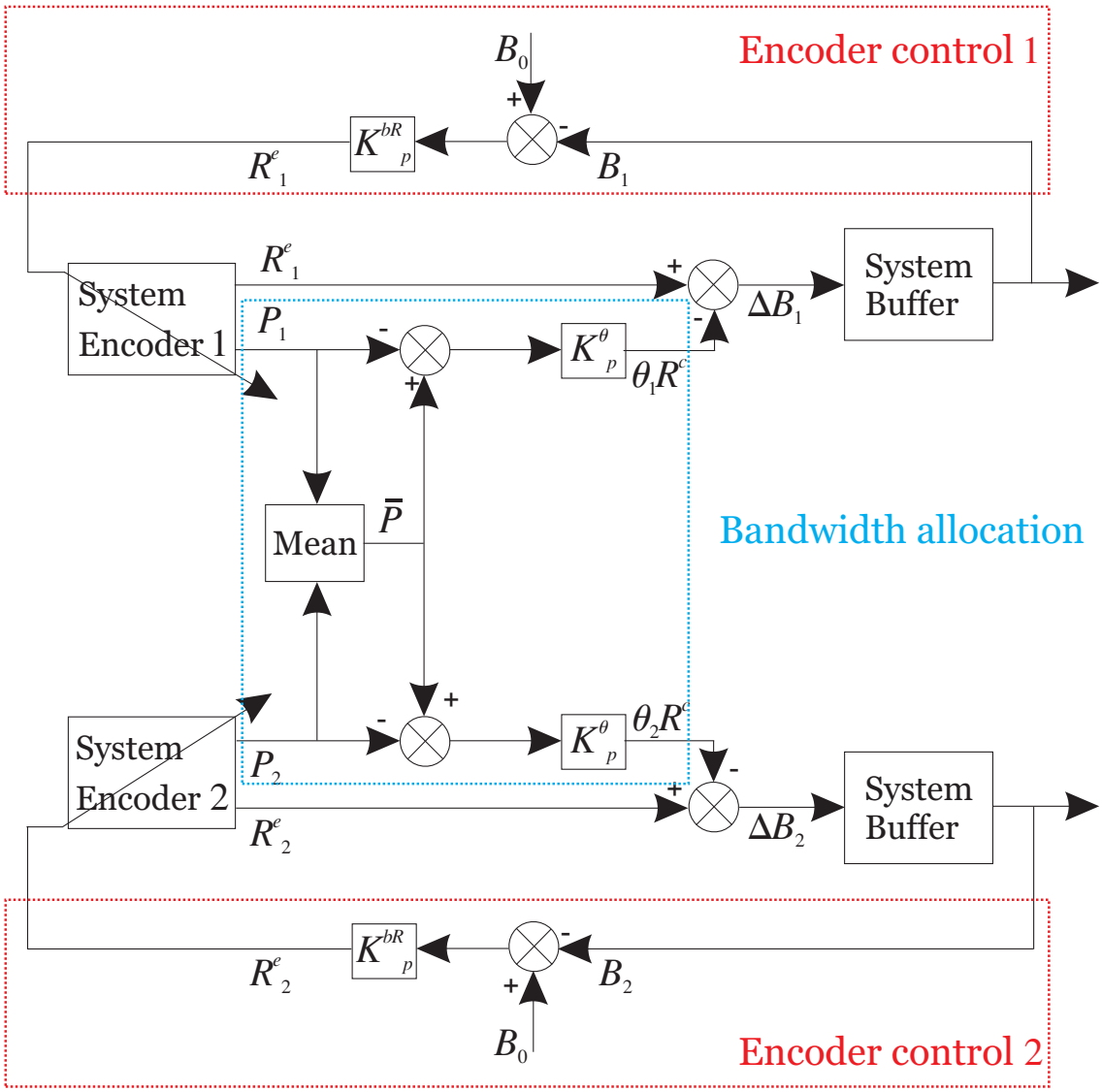


Figure 5.3: Details of the system and control blocks in the closed-loop control system, when buffer levels in bits are controlled and two programs are multiplexed.

We assume that each video server i provides to the MANE the PSNR $P_i(j-1)$ of the $j-1$-th encoded VU. The average PSNR $\bar{P}(j-1)$ among the N programs is then computed.

At time j, the centralized bandwidth allocation control requires the PSNR values of the last encoded VU of the N programs $P_{i=1\ldots N}(j-1)$ and the reference rate $R_0 = R^c/N$. In fact, at time j, the control process in the MANE use the last available PSNR values, and due to the encoding time at the encoders, a delay between the control time at the MANE and the PSNR values occurs.

The transmission rate $R^{\mathrm{t}}_i(j) = \theta_i(j)R^{\mathrm{c}}$ allocated to the i-th program between time $j-1$ and time j is evaluated using the reference encoding rate $R_0 = R^{\mathrm{c}}/N$.

Using a PI controller, the transmission rate at time j are

$$R_i^{\mathrm{t}}(j) = \theta_i(j) R^{\mathrm{c}} = \frac{R^{\mathrm{c}}}{N} + K_{\mathrm{P}}^{\theta}(\bar{P}(j-1) - P_i(j-1)) + K_{\mathrm{I}}^{\theta} \sum_{k=1}^{j-1} (\bar{P}(k) - P_i(k)), \quad (5.2)$$

where K_{P}^{θ} and K_{I}^{θ} are the proportional and the integral correction gain expressed in (bits dB^{-1} s^{-1}) to ensure homogeneity, and $\bar{P}(j-1) = \frac{1}{N} \sum_{k=1}^{N} P_k(j-1)$.

More (less) rate is allocated to programs with a PSNR smaller (larger) than $\bar{P}(j-1)$. Thus, according to (5.1), the buffer level of programs which produces an encoded video with smaller PSNR than the average PSNR $\bar{P}(j-1)$ decreases faster than those of programs producing video with a larger PSNR. For those programs, the buffer level may even decrease.

3.2 Distributed encoder control

For each video encoder an individual rate control is performed for each VU. The encoding rate can be adjusted depending on the buffer levels in bits or on the buffer levels in seconds according to a reference coding rate $R_0 = R^{\mathrm{c}}/N$ depending on system requirements.

3.2.1 Buffer levels in bits

The encoding rate for the j-th VU of each video program is controlled to limit deviations of the level in bits $B_i(j)$ from the reference buffer level in bits B_0 (this prevents buffer under and overflow).

At time j, the i-th encoder control requires the buffer level in bits $B_i\,(j)$ associated to only the i-th program, the reference buffer level in bits B_0, and the channel rate R^{c}. Here, we assume that the buffer level in bits $B_i\,(j)$ calculated in the centralized bandwidth allocation is fed back, without delay, to its associated encoder control via a wire-line link.

Using a PI controller, the encoding rate for the j-th VU are taken as

$$R_i^{\mathrm{e}}(j) = \tfrac{R^{\mathrm{c}}}{N} - K_{\mathrm{P}}^{\mathrm{bR}} \left(\tfrac{B_i(j) - B_0}{T} \right) - K_{\mathrm{I}}^{\mathrm{bR}} \textstyle\sum_{k=1}^{j} \left(\tfrac{B_i(k) - B_0}{T} \right), \quad (5.3)$$

where $K_{\mathrm{P}}^{\mathrm{bR}}$ and $K_{\mathrm{I}}^{\mathrm{bR}}$ are the (dimensionless) proportional and integral correction factors. The encoding rate increases (decreases) when the buffer is below (above) the reference level B_0.

Combining (5.2) and (5.3), we see that buffers corresponding to programs producing encoded video with smaller PSNR than the average PSNR $\bar{P}(j-1)$ are drained faster. As a consequence, the encoding rate allowed to encode the next VU is increased. This means that the encoding rate is increased, or decreased, according to the PSNR of the already encoded VUs.

This control technique only requires the MANE to feed back the buffer level in bits $B_i(j)$ to the i-th remote video encoder. This feedback with non-congested network conditions takes about tens of milliseconds, which is negligible compared to the time constants of video coders when the control is performed at the GoP level. The feedback delay is then neglected.

3.2.2 Buffer levels in seconds

The encoder control may also target a similar buffering delay between the N programs. This constraint is very important to consider in delay sensitive applications. In this approach, the encoding rate should be adjusted to limit deviations of the buffer level in seconds $\tau_i(j)$ from some reference buffer level τ_0.

Here, at time j, the i-th encoder control requires the estimated buffer level in second $\tilde{\tau}_i\,(j)$, associated only to the i-th program, the reference buffer level τ_0, and the channel rate R^c. Here as in the control of buffer level in bits, we assume that the buffer level in seconds $\tau_i\,(j)$ calculated in the centralized bandwidth allocation is immediately fed back without delay.

The buffer levels in seconds in the i-th buffer is

$$\tau_i(j) = f_i(j)T \tag{5.4}$$

where $f_i(j)$ is the number of VUs in the i-th buffer (in the MANE) at time j. It can also be obtained from the buffer level in bits $B_i(j)$ as

$$\tau_i(j) = \frac{B_i(j)}{\bar{R}_i^{\mathrm{e}}(j)}, \tag{5.5}$$

where $\bar{R}_i^{\mathrm{e}}(j)$ is the average encoding rate of the already buffered VUs in buffer i at time j obtained by

$$\bar{R}_i^{\mathrm{e}}(j) = \sum_{l=1}^{\lfloor f_i(j)\rfloor} R_i^{\mathrm{e}}(j) + R_i^{\mathrm{e}}(j - \lceil f_i(j)\rceil)\,(f_i(j) - \lfloor f_i(j)\rfloor)\,. \tag{5.6}$$

Here, $\lfloor . \rfloor$ denotes rounding towards $-\infty$, and $\lceil . \rceil$ towards $+\infty$. Nevertheless, $f_i(j)$ is not easy to estimate due to the fine granularity of the draining rate. The average encoding rate $\bar{R}_i^{\mathrm{e}}(j)$ may be approximated as follows

$$\begin{aligned} \tilde{R}_i^{\,\mathrm{e}}(1) &= R_i^{\mathrm{e}}(1) \\ \tilde{R}_i^{\,\mathrm{e}}(j) &= \alpha R_i^{\mathrm{e}}(j-1) + (1-\alpha)\tilde{R}_i^{\,\mathrm{e}}(j-1),\ j > 1 \end{aligned} \tag{5.7}$$

where $0 < \alpha < 1$ is some forgetting factor. The estimated buffer level in seconds $\tilde{\tau}_i(j)$ is

$$\tilde{\tau}_i(j) = \frac{B_i(j)}{\tilde{R}_i^{\mathrm{e}}(j)}. \tag{5.8}$$

The PI control of the encoding rate $R_i^{\mathrm{e}}(j)$ for the j-th VU between time $j-1$ and time j, is performed as follows

$$R_i^{\mathrm{e}}(j) = R_0 - K_{\mathrm{P}}^{\tau\mathrm{R}} \left(\frac{\tilde{\tau}_i(j) - \tau_0}{T} \right) - K_{\mathrm{I}}^{\tau\mathrm{R}} \sum_{k=1}^{j} \left(\frac{\tilde{\tau}_i(k) - \tau_0}{T} \right), \tag{5.9}$$

where the proportional and the integral correction factors $K_{\mathrm{P}}^{\tau\mathrm{R}}$ and $K_{\mathrm{I}}^{\tau\mathrm{R}}$ are expressed in bits s^{-1}. From (5.9) and (5.2), we notice that the encoding rate increases when the buffer level in seconds is below the reference level τ_0. Since the buffers corresponding to programs producing encoded video with PSNR smaller than the average PSNR $\bar{P}(j-1)$ are drained faster, the encoding rate is increased to reach the reference delay. Thus, the encoding rate is increased or decreased according to the PSNR of the already encoded VUs.

4 Equilibrium and stability analysis

In this section, we study the equilibrium and the stability of the proposed bandwidth allocation and encoder rate control. The proposed closed-loop feedback control system should be stable to meet the above-mentioned performance objective. Thus, the stability of the system is studied considering N sources assuming random variations of the source characteristics.

4.1 Source model

Consider the following PSNR-Rate model for video source

$$P_i(j) = \beta_i(j) + \gamma_i(j) T R_i^{\mathrm{e}}(j), \tag{5.10}$$

where $\beta_i(j)$ and $\gamma_i(j)$ are parameters that depend on the VU j and the video program i. The parameters $\beta_i(j)$ and $\gamma_i(j)$ can be modeled by

$$\beta_i(j) = \beta_i(j) + \delta\beta_i(j), \tag{5.11}$$

and

$$\gamma_i(j) = \gamma_i(j) + \delta\gamma_i(j), \tag{5.12}$$

where $\delta\beta_i\,(j)$ and $\delta\gamma_i\,(j)$ represent the variations of the source characteristics. $\beta_i\,(j)$ and $\gamma_i\,(j)$ can be estimated for each VU using two encoding trials. The PSNR-Rate model in (5.10) can be used for encoded video or Gaussian sources.

In fact, consider N Gaussian sources with mean 128 and variance $\sigma_i^2(j)$ with a single sample provided by each source. The classical Rate-Distortion function is

$$D_i(j)(R_i^{\mathrm{e}}(j)T) = \sigma_i(j)^2 2^{-2R_i^{\mathrm{e}}(j)T} \tag{5.13}$$

with $i = 1, \ldots, N$. The variance $\sigma_i(j)$ is such that samples are very likely to remain within $[0, 255\,[$. Thus, the theoretical PSNR is

$$P_i(j) = 10\log_{10}\left(\frac{255^2}{{\sigma_i}^2(j)}\right) + 6.02 T R_i^{\mathrm{e}}(j) = \beta_i(j) + \gamma T R_i^{\mathrm{e}}(j), \tag{5.14}$$

with $\beta_i(j) = 10\log_{10}(\frac{255^2}{{\sigma_i}^2(j)})$ and $\gamma = 6.02$ dB/bits.

4.2 Control of the buffer levels in bits

Two controllers are considered in the proposed control system.

- First, an individual distributed control system is performed at each encoder using a PI controller to determine the encoding rate and requiring the buffer levels in bits $B_i(j)$ associated to each program $i = 1 \ldots N$, the reference buffer level in bits B_0 and the reference rate R_0.
- Second, a centralized bandwidth allocation control is performed to adjust the allocated bandwidth among programs, using a PI controller requiring PSNR values of the last encoded VU of the N programs $P_{i=1\ldots N}\,(j-1)$ and the reference rate R_0.

The buffer level evolution defined by

$$B_i(j+1) = B_i(j) + \left(R_i^e(j) - R_i^t(j)\right) T \tag{5.15}$$

involves the encoding rate provided by the distributed encoder control process and the transmission rate provided by the centralized bandwidth allocation control process.

The transmission rate in (5.2) is rewritten as

$$\begin{aligned} R_i^t(j) &= R_0 + K_P^\theta \left(\frac{1}{N} \sum_{l=1}^{N} P_i(j-1) - P_i(j-1) \right) \\ &\quad + K_I^\theta \left(\phi_i(j) + \left(\frac{1}{N} \sum_{l=1}^{N} P_l(j-1) - P_i(j-1) \right) \right), \end{aligned} \tag{5.16}$$

where $\phi_i(j)$ is the cumulated PSNR discrepancy defined as follows

$$\phi_i(j+1) = \phi_i(j) + \left(\frac{1}{N} \sum_{l=1}^{N} P_l(j-1) - P_i(j-1) \right), \tag{5.17}$$

this means that PSNR values $P_{i=1\ldots N}(j-1)$ are assumed available at time j at the bandwidth allocation process.

Consider the new state

$$\tilde{P}_i(j+1) = P_i(j), \tag{5.18}$$

thus, the transmission rate can be expressed as follows

$$R_i^t(j) = R_0 + \left(K_P^\theta + K_I^\theta\right) \left(\frac{1}{N} \sum_{l=1}^{N} \tilde{P}_l(j) - \tilde{P}_i(j) \right) + K_I^\theta \phi_i(j). \tag{5.19}$$

The encoding rate in (5.3) is rewritten as

$$R_i^e(j) = R_0 - \frac{K_P^{\mathrm{bR}}}{T} \left(B_i(j) - B_0\right) - \frac{K_I^{\mathrm{bR}}}{T} \left(\Delta_i(j) + \left(B_i(j) - B_0\right)\right), \tag{5.20}$$

where $\Delta_i(j)$ is the cumulated buffer discrepancy defined as follows

$$\Delta_i(j+1) = \Delta_i(j) + \left(B_i(j) - B_0\right), \tag{5.21}$$

this means that at time j the buffer level in bits $B_i(j)$ is assumed immediately available at the encoder control process.

Consider the following change of variable

$$\tilde{B}_i(j) = B_i(j) - B_0, \tag{5.22}$$

thus, the encoding rate can be expressed by

$$R_i^e(j) = R_0 - \frac{K_P^{\mathrm{bR}} + K_I^{\mathrm{bR}}}{T} \tilde{B}_i(j) - \frac{K_I^{\mathrm{bR}}}{T} \Delta_i(j). \tag{5.23}$$

Thus, the following state-space representation describing the interconnection between the various measurable properties of the system may be introduced using (5.10), (5.11),(5.15), (5.17),(5.19), (5.21), and (5.23)

$$\left\{\begin{array}{rcl}\beta_i(j+1) &=& \beta_i(j)+\delta\beta_i(j)\\ \Delta_i(j+1) &=& \Delta_i(j)+\tilde{B}_i(j)\\ \phi_i(j+1) &=& \phi_i(j)+\left(\frac{1}{N}\sum_{l=1}^{N}\tilde{P}_l(j)\right)-\tilde{P}_i(j)\\ \tilde{P}_i(j+1) &=& \beta_i(j)+\gamma_i(j)\left(R_0-\left(K_P^{\mathrm{bR}}-K_I^{\mathrm{bR}}\right)\tilde{B}_i(j)+K_I^{\mathrm{bR}}\Delta_i(j)\right)T\\ \tilde{B}_i(j+1) &=& \tilde{B}_i(j)-\left(\left(K_P^{\mathrm{bR}}+K_I^{\mathrm{bR}}\right)\tilde{B}_i(j)+K_I^{\mathrm{bR}}\Delta_i(j)\right)\\ &-& \left(\left(K_P^{\theta}+K_I^{\theta}\right)\left(\frac{1}{N}\sum_{l=1}^{N}\tilde{P}_l(j)-\tilde{P}_i(j)\right)+K_I^{\theta}\phi_i(j)\right)T\end{array}\right. \tag{5.24}$$

In what follows, we assume that $\gamma_i(j)$ remains constant so that $\gamma_i(j)=\gamma$. Thus, one gets

$$\tilde{\beta}_i(j)=\beta_i(j)+\gamma T R_0. \tag{5.25}$$

4.2.1 Equilibrium analysis

The equilibrium of (5.26) is obtained when

$$\left\{\begin{array}{rcl}\tilde{\beta}_i^{\mathrm{eq}} &=& \tilde{\beta}_i^{\mathrm{eq}}+\delta\beta_i^{\mathrm{eq}}\\ \Delta_i^{\mathrm{eq}} &=& \Delta_i^{\mathrm{eq}}+\tilde{B}_i^{\mathrm{eq}}\\ \phi_i^{\mathrm{eq}} &=& \phi_i^{\mathrm{eq}}+\left(\left(\frac{1}{N}\sum_{l=1}^{N}\tilde{P}_l^{\mathrm{eq}}\right)-\tilde{P}_i^{\mathrm{eq}}\right.\\ \tilde{P}_i^{\mathrm{eq}} &=& \tilde{\beta}_i^{\mathrm{eq}}-\gamma\left(\left(K_P^{\mathrm{bR}}+K_I^{\mathrm{bR}}\right)\tilde{B}_i^{\mathrm{eq}}+K_I^{\mathrm{bR}}\Delta_i^{\mathrm{eq}}\right)\\ \tilde{B}_i^{\mathrm{eq}} &=& \tilde{B}_i^{\mathrm{eq}}-\left(\left(K_P^{\mathrm{bR}}+K_I^{\mathrm{bR}}\right)\tilde{B}_i^{\mathrm{eq}})+K_I^{\mathrm{bR}}\Delta_i^{\mathrm{eq}}\right)\\ &-& \left(\left(K_P^{\theta}+K_I^{\theta}\right)\left(\frac{1}{N}\sum_{l=1}^{N}\tilde{P}_l^{\mathrm{eq}}-\tilde{P}_i^{\mathrm{eq}}\right)+K_I^{\theta}\phi_i^{\mathrm{eq}}\right)T\end{array}\right., \tag{5.26}$$

which leads to

$$\begin{array}{rcl}\delta\beta_i^{\mathrm{eq}} &=& 0\\ \tilde{B}_i^{\mathrm{eq}} &=& 0\\ \tilde{P}_i^{\mathrm{eq}}-\frac{1}{N}\sum_{k=1}^{N}\tilde{P}_k^{\mathrm{eq}} &=& 0\\ \tilde{P}_i^{\mathrm{eq}}-\beta_i^{\mathrm{eq}}-\gamma K_I^{bR}\Delta_i^{\mathrm{eq}} &=& 0\\ \left(-\frac{K_I^{bR}}{T}\Delta_i^{\mathrm{eq}}+K_I^{\theta}\phi_i^{\mathrm{eq}}\right)T &=& 0\end{array} \tag{5.27}$$

This is a system of five equations with five unknowns, since all β_i^{eq} are known source characteristics.

The system resolution in (5.27) can be rewritten as

$$\begin{pmatrix} I & 0 & 0 & 0 & 0\\ 0 & I & 0 & 0 & 0\\ 0 & 0 & L & 0 & 0\\ 0 & 0 & I & -\gamma K_I^{bR} I & 0\\ 0 & 0 & 0 & -K_I^{bR} I & K_I^{\theta} T I\end{pmatrix}\begin{pmatrix}\delta\beta_i^{\mathrm{eq}}\\ \tilde{B}_i^{\mathrm{eq}}\\ \tilde{P}_i^{\mathrm{eq}}\\ \Delta_i^{\mathrm{eq}}\\ \phi_i^{\mathrm{eq}}\end{pmatrix}=\begin{pmatrix}0\\0\\0\\ \tilde{\beta}_i^{\mathrm{eq}}\\0\end{pmatrix}, \tag{5.28}$$

with

$$L=\frac{1}{N}\begin{pmatrix}N-1 & -1 & -1\\ -1 & \ddots & -1\\ -1 & -1 & N-1\end{pmatrix}.$$

The matrix

$$\begin{pmatrix} I & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & L & 0 & 0 \\ 0 & 0 & I & -\gamma K_I^{bR} I & 0 \\ 0 & 0 & 0 & -K_I^{bR} I & K_I^{\theta} T I \end{pmatrix}$$

is a full rang matrix when $K_I^{bR} \neq 0$ and $K_I^{\theta} \neq 0$, thus, solutions of the system resolution in (5.27) using PI controllers exit.

At equilibrium, $\tilde{P}_i^{\text{eq}} = \frac{1}{N} \sum_{k=1}^{N} \tilde{P}_k^{\text{eq}}$, $i \in \{1, \dots N\}$, quality fairness among the N programs is satisfied. In addition, at equilibrium, from $\tilde{B}_i^{\text{eq}} = 0$, the target reference buffer level in bits is reached.

4.2.2 Stability

The stability analysis of the proposed system involves the following discrete-time state-space representation as follows

$$\begin{aligned} X^{\text{b}}(j+1) &= A^{\text{b}} X^{\text{b}}(j) + B^{\text{b}} U^{\text{b}}(j) \\ Y^{\text{b}}(j) &= C^{\text{b}} X^{\text{b}}(j), \end{aligned} \tag{5.29}$$

where system state X^{b} and input U^{b} vectors

$$\begin{aligned} X^{\text{b}}(j) = \Big(& \tilde{\beta}_1(j), \dots, \tilde{\beta}_N(j), \Delta_1(j), \dots, \Delta_N(j), \phi_1(j), \dots, \phi_N(j), \\ & \tilde{P}_1(j), \dots, \tilde{P}_N(j), \tilde{B}_1(j), \dots, \tilde{B}_N(j) \Big)^T, \end{aligned} \tag{5.30}$$

and

$$U^{\text{b}}(j) = \left(\delta\beta_1(j), \dots, \delta\beta_N(j) \right)^T. \tag{5.31}$$

In the considered system, we assume that only the state of the buffer and the PSNR of the encoded VUs may be measured. Thus in (5.29), one has

$$\mathrm{A}^{\text{b}} = \begin{pmatrix} I & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & I \\ 0 & 0 & I & -L & 0 \\ I & -\gamma K_I^{bR} I & 0 & 0 & -\gamma \left(K_P^{bR} + K_I^{bR} \right) I \\ 0 & -K_I^{R} I & -K_I^{\theta} T I & \left(K_P^{\theta} + K_I^{\theta} \right) T L & \left(1 - \left(K_P^{bR} + K_I^{bR} \right) \right) I \end{pmatrix}, \tag{5.32}$$

$$B^{\text{b}} = \begin{pmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{5.33}$$

$$C^{\text{b}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{pmatrix}, \tag{5.34}$$

with

$$L = \frac{1}{N}\begin{pmatrix} N-1 & -1 & -1 \\ -1 & \ddots & -1 \\ -1 & -1 & N-1 \end{pmatrix}.$$

When the bandwidth allocation and the encoding rate control are performed using PI controllers, after some derivations (Appendix B), we get

$$\det\left(zI - A^{\mathrm{b}}\right) = (z-1)^N \left(a_1 z^4 + a_2 z^3 + a_3 z^2 + a_4 z + a_5\right)^N \tag{5.35}$$

with

$$\begin{cases} a_1 &= 1 \\ a_2 &= -3 + \left(K_P^{bR} + K_I^{bR}\right) \\ a_3 &= 3 - 2\left(K_P^{bR} + K_I^{bR}\right) + K_I^{bR} + \gamma\left(K_P^{bR} + K_I^{bR}\right)\left(K_P^{\theta} + K_I^{\theta}\right)T \\ a_4 &= -1 + K_P^{bR} - 2\gamma K_P^{bR}\left(K_P^{\theta} + K_I^{\theta}\right)T + \gamma K_P^{bR}\left(K_I^{\theta} + K_P^{\theta}\right)T + \gamma K_I^{bR} K_P^{\theta} T \\ a_5 &= \gamma K_I^{bR} K_I^{\theta} T - \gamma K_I^{bR}\left(K_P^{\theta} + K_I^{\theta}\right)T + \gamma\left(K_P^{bR} + K_I^{bR}\right)K_P^{\theta} T \end{cases}. \tag{5.36}$$

The locations of the roots of $\det\left(zI - A^{\mathrm{b}}\right)$ with respect to the unit circle determine the stability of the system. The root $z = 1$ is located on the unit circle and is related to the dynamic of the source defined in (5.11). For the second part of (5.35), the roots can be obtained numerically [Nic09] allowing to predict whether a given choice of K_P^{θ}, K_I^{θ}, K_P^{bR}, and K_I^{bR} will lead to a stable system.

4.3 Control of the buffer levels in seconds

Here, as when the buffer levels in bits are controlled, two controllers are considered in the proposed control system.

- First, an individual distributed control system is performed at each encoder using a PI controller to determine the encoding rate and requiring the estimated buffer levels in seconds $\tilde{\tau}_i(j)$ associated to each program $i = 1 \ldots N$, the reference buffer level in seconds τ_0 and the reference rate R_0.

- Second, a centralized bandwidth allocation control is performed to adjust the allocated bandwidth among programs, using a PI controller requiring PSNR values of the last encoded VU of the N programs $P_{i=1\ldots N}(j-1)$ and the reference rate R_0.

As in the previous control case, the transmission rate is defined by

$$R_i^t(j) = R_0 + \left(K_P^{\theta} + K_I^{\theta}\right)\left(\frac{1}{N}\sum_{l=1}^{N}\tilde{P}_l(j) - \tilde{P}_i(j)\right) + K_I^{\theta}\phi_i(j). \tag{5.37}$$

The estimated buffer level in seconds $\tilde{\tau}_i(j)$ is defined by

$$\tilde{\tau}_i(j) = \frac{B_i(j)}{\tilde{R}_i^{\mathrm{e}}(j)}, \tag{5.38}$$

where $\tilde{R}_i^{\mathrm{e}}(j)$ is the estimated average encoding rate of the already buffered VUs in buffer i at time j obtained by

$$\begin{aligned} \tilde{R}_i{}^{\mathrm{e}}(1) &= R_i^{\mathrm{e}}(1) \\ \tilde{R}_i{}^{\mathrm{e}}(j) &= \alpha R_i^{\mathrm{e}}(j-1) + (1-\alpha)\tilde{R}_i{}^{\mathrm{e}}(j-1), \end{aligned} \tag{5.39}$$

where $\alpha < 1$ is some forgetting factor.

The encoding rate at each encoder control in (5.9) performed according to the estimated buffer level in seconds $\tilde{\tau}_i(j)$ can be rewritten as follows

$$R_i^{\mathrm{e}}(j) = R_0 - \frac{K_P^{\tau R}}{T}\left(\tilde{\tau}_i(j) - \tau_0\right) - \frac{K_I^{\tau R}}{T}\left(\Pi_i(j) + \left(\tilde{\tau}_i(j) - \tau_0\right)\right) \tag{5.40}$$

where $\Pi_i(j)$ is the cumulated discrepancies of the buffer level in seconds

$$\Pi_i(j+1) = \Pi_i(j) + \left(\tilde{\tau}_i(j) - \tau_0\right). \tag{5.41}$$

One sees that (5.41) imposes that $\tilde{\tau}_i(j)$ is immediately available at the encoder control at time j.

When the encoding rate targets similar buffer levels in seconds and when PI controllers are used in both the bandwidth allocation and the encoding rate control, the system may be described by the following discrete-time nonlinear state-space equation

$$\left\{\begin{array}{rcl} \tilde{\beta}_i(j+1) &=& \tilde{\beta}_i(j) + \delta\beta_i(j) \\ \phi_i(j+1) &=& \phi_i(j) + \frac{1}{N}\left(\sum_{k=1}^{N}\tilde{P}_k(j)\right) - \tilde{P}_i(j) \\ \Pi_i(j+1) &=& \Pi_i(j) + \left(\frac{B_i(j)}{\tilde{R}_i^{\mathrm{e}}(j)} - \tau_0\right) \\ \tilde{R}_i^{\mathrm{e}}(j+1) &=& \alpha\left(R_0 - \frac{K_P^{\tau R}+K_I^{\tau R}}{T}\left(\frac{B_i(j)}{\tilde{R}_i^{\mathrm{e}}(j)} - \tau_0\right) - \frac{K_I^{\tau R}}{T}\Pi_i(j)\right) + (1-\alpha)\tilde{R}_i^{\mathrm{e}}(j) \\ \tilde{P}_i(j+1) &=& \tilde{\beta}_i(j) - \gamma\left(\frac{K_P^{\tau R}+K_I^{\tau R}}{T}\left(\frac{B_i(j)}{\tilde{R}_i^{\mathrm{e}}(j)} - \tau_0\right) - \frac{K_I^{\tau R}}{T}\Pi_i(j)\right)T \\ B_i(j+1) &=& B_i(j) + \left(R_0 - \frac{K_P^{\tau R}+K_I^{\tau R}}{T}\left(\frac{B_i(j)}{\tilde{R}_i^{\mathrm{e}}(j)} - \tau_0\right) - \frac{K_I^{\tau R}}{T}\Pi_i(j)\right)T \\ && - \left(R_0 + \left(K_P^{\theta}+K_I^{\theta}\right)\left(\frac{1}{N}\sum_{k=1}^{N}\tilde{P}_k(j) - \tilde{P}_i(j)\right) + K_I^{\theta}\phi_i(j)\right)T \end{array}\right., \tag{5.42}$$

where $\phi_i(j)$ and $\Pi_i(j)$ are respectively the cumulated PSNR and delay discrepancies. The stability analysis of (5.45) requires its linearization around some equilibrium.

$$\left\{\begin{array}{rcl} \Delta\tilde{\beta}_i(j+1) &=& \frac{\partial u}{\partial\tilde{\beta}_i}|_{\mathrm{op}}\Delta\tilde{\beta}_i(j) + \frac{\partial u}{\partial\tilde{R}_i^{\mathrm{e}}}|_{\mathrm{op}}\Delta\tilde{R}_i^{\mathrm{e}}(j) + \frac{\partial u}{\partial\Pi_i}|_{\mathrm{op}}\Delta\Pi_i(j) + \frac{\partial u}{\partial\phi_i}|_{\mathrm{op}}\Delta\phi_i(j) + \frac{\partial u}{\partial\tilde{P}_i}|_{\mathrm{op}}\Delta\tilde{P}_i(j) + \frac{\partial u}{\partial B_i}|_{\mathrm{op}}\Delta B_i(j) \\ \Delta\tilde{R}_i^{\mathrm{e}}(j+1) &=& \frac{\partial v}{\partial\tilde{\beta}_i}|_{\mathrm{op}}\Delta\tilde{\beta}_i(j) + \frac{\partial v}{\partial\tilde{R}_i^{\mathrm{e}}}|_{\mathrm{op}}\Delta\tilde{R}_i^{\mathrm{e}}(j) + \frac{\partial v}{\partial\Pi_i}|_{\mathrm{op}}\Delta\Pi_i(j) + \frac{\partial v}{\partial\phi_i}|_{\mathrm{op}}\Delta\phi_i(j) + \frac{\partial v}{\partial\tilde{P}_i}|_{\mathrm{op}}\Delta\tilde{P}_i(j) + \frac{\partial v}{\partial B_i}|_{\mathrm{op}}\Delta B_i(j) \\ \Delta\Pi_i(j+1) &=& \frac{\partial w}{\partial\tilde{\beta}_i}|_{\mathrm{op}}\Delta\tilde{\beta}_i(j) + \frac{\partial w}{\partial\tilde{R}_i^{\mathrm{e}}}|_{\mathrm{op}}\Delta\tilde{R}_i^{\mathrm{e}}(j) + \frac{\partial w}{\partial\Pi_i}|_{\mathrm{op}}\Delta\Pi_i(j) + \frac{\partial w}{\partial\phi_i}|_{\mathrm{op}}\Delta\phi_i(j) + \frac{\partial w}{\partial\tilde{P}_i}|_{\mathrm{op}}\Delta\tilde{P}_i(j) + \frac{\partial w}{\partial B_i}|_{\mathrm{op}}\Delta B_i(j) \\ \Delta\phi_i(j+1) &=& \frac{\partial x}{\partial\tilde{\beta}_i}|_{\mathrm{op}}\Delta\tilde{\beta}_i(j) + \frac{\partial x}{\partial\tilde{R}_i^{\mathrm{e}}}|_{\mathrm{op}}\Delta\tilde{R}_i^{\mathrm{e}}(j) + \frac{\partial x}{\partial\Pi_i}|_{\mathrm{op}}\Delta\Pi_i(j) + \frac{\partial x}{\partial\phi_i}|_{\mathrm{op}}\Delta\phi_i(j) + \frac{\partial x}{\partial\tilde{P}_i}|_{\mathrm{op}}\Delta\tilde{P}_i(j) + \frac{\partial x}{\partial B_i}|_{\mathrm{op}}\Delta B_i(j) \\ \Delta\tilde{P}_i(j+1) &=& \frac{\partial y}{\partial\tilde{\beta}_i}|_{\mathrm{op}}\Delta\tilde{\beta}_i(j) + \frac{\partial y}{\partial\tilde{R}_i^{\mathrm{e}}}|_{\mathrm{op}}\Delta\tilde{R}_i^{\mathrm{e}}(j) + \frac{\partial y}{\partial\Pi_i}|_{\mathrm{op}}\Delta\Pi_i(j) + \frac{\partial y}{\partial\phi_i}|_{\mathrm{op}}\Delta\phi_i(j) + \frac{\partial y}{\partial\tilde{P}_i}|_{\mathrm{op}}\Delta\tilde{P}_i(j) + \frac{\partial y}{\partial B_i}|_{\mathrm{op}}\Delta B_i(j) \\ \Delta B_i(j+1) &=& \frac{\partial z}{\partial\tilde{\beta}_i}|_{\mathrm{op}}\Delta\tilde{\beta}_i(j) + \frac{\partial z}{\partial\tilde{R}_i^{\mathrm{e}}}|_{\mathrm{op}}\Delta\tilde{R}_i^{\mathrm{e}}(j) + \frac{\partial z}{\partial\Pi_i}|_{\mathrm{op}}\Delta\Pi_i(j) + \frac{\partial z}{\partial\phi_i}|_{\mathrm{op}}\Delta\phi_i(j) + \frac{\partial z}{\partial\tilde{P}_i}|_{\mathrm{op}}\Delta\tilde{P}_i(j) + \frac{\partial z}{\partial B_i}|_{\mathrm{op}}\Delta B_i(j) \end{array}\right., \tag{5.43}$$

where $u(.)$, $v(.)$, $w(.)$, $x(.)$, $y(.)$ and $z(.)$ are non linear functions, and the subindex op is for *operating point*, which is typically an equilibrium of the system. This leads to

$$\begin{cases} \Delta\tilde{\beta}_i\left(j+1\right) = & \Delta\tilde{\beta}_i\left(j\right) \\ \Delta\phi_i\left(j+1\right) = & \Delta\phi_i\left(j\right) + \frac{1}{N}\left(\sum_{k=1}^{N}\Delta\tilde{P}_k\left(j\right)\right) - \Delta\tilde{P}_i\left(j\right) \\ \Delta\Pi_i\left(j+1\right) = & \Delta\Pi_i\left(j\right) - \left(\frac{B_i^{\mathrm{op}}}{\left(\tilde{R}_i^{e\,\mathrm{op}}\right)^2}\right)\Delta\tilde{R}_i^{e}(j) + \frac{\Delta B_i(j)}{\tilde{R}_i^{e\,\mathrm{op}}} \\ \Delta\tilde{R}_i^{\mathrm{e}}(j+1) = & -\alpha\left(\frac{K_P^{\tau R}}{T} + \frac{K_I^{\tau R}}{T}\right)\left(-\frac{B_i^{\mathrm{op}}}{\left(\tilde{R}_i^{e\,\mathrm{op}}\right)^2}\Delta\tilde{R}_i^{e}(j) + \frac{\Delta B_i(j)}{\tilde{R}_i^{e\,\mathrm{op}}}\right) - \alpha\frac{K_I^{\tau R}}{T}\Delta\Pi_i\left(j\right) + (1-\alpha)\Delta\tilde{R}_i^{\mathrm{e}}(j) \\ \Delta\tilde{P}_i\left(j+1\right) = & \Delta\tilde{\beta}_i\left(j+1\right) - \gamma\left(\left(K_P^{\tau R} + K_I^{\tau R}\right)\left(-\frac{B_i^{\mathrm{op}}}{\left(\tilde{R}_i^{e\,\mathrm{op}}\right)^2}\Delta\tilde{R}_i^{e}(j) + \frac{\Delta B_i(j)}{\tilde{R}_i^{e\,\mathrm{op}}}\right) + K_I^{\tau R}\Delta\Pi_i\left(j\right)\right) \\ \Delta B_i\left(j+1\right) = & \Delta B_i(j) - \left(K_P^{\tau R} + K_I^{\tau R}\right)\left(-\frac{B_i(j)}{\left(\tilde{R}_i^{e\,\mathrm{op}}\right)^2}\Delta\tilde{R}_i^{e\,\mathrm{op}} + \frac{\Delta B_i(j)}{\tilde{R}_i^{e\,\mathrm{op}}}\right) - K_I^{\tau R}\Delta\Pi_i\left(j\right) \\ & - \left(\left(K_P^{\theta} + K_I^{\theta}\right)\left(\frac{1}{N}\sum_{k=1}^{N}\Delta\tilde{P}_k\left(j\right) + \Delta\tilde{P}_i\left(j\right)\right) + K_I^{\theta}\Delta\phi_i\left(j\right)\right)T \end{cases} \tag{5.44}$$

4.3.1 Equilibrium

At equilibrium

$$\begin{cases} \tilde{\beta}_i^{\mathrm{eq}} &= \tilde{\beta}_i^{\mathrm{eq}} + \delta\beta_i^{\mathrm{eq}} \\ \phi_i^{\mathrm{eq}} &= \phi_i^{\mathrm{eq}} + \frac{1}{N}\left(\sum_{k=1}^{N}\tilde{P}_k^{\mathrm{eq}}\right) - \tilde{P}_i^{\mathrm{eq}} \\ \Pi_i^{\mathrm{eq}} &= \Pi_i^{\mathrm{eq}} + \left(\frac{B_i^{\mathrm{eq}}}{\tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}}} - \tau_0\right) \\ \tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}} &= \alpha\left(R_0 - \frac{K_P^{\tau R}+K_I^{\tau R}}{T}\left(\frac{B_i^{\mathrm{eq}}}{\tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}}} - \tau_0\right) - \frac{K_I^{\tau R}}{T}\Pi_i^{\mathrm{eq}}\right) + (1-\alpha)\tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}} \\ \tilde{P}_i^{\mathrm{eq}} &= \tilde{\beta}_i^{\mathrm{eq}} - \gamma\left(\frac{K_P^{\tau R}+K_I^{\tau R}}{T}\left(\frac{B_i^{\mathrm{eq}}}{\tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}}} - \tau_0\right) - \frac{K_I^{\tau R}}{T}\Pi_i\left(j\right)\right)T \\ B_i^{\mathrm{eq}} &= B_i^{\mathrm{eq}} + \left(R_0 - \frac{K_P^{\tau R}+K_I^{\tau R}}{T}\left(\frac{B_i^{\mathrm{eq}}}{\tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}}} - \tau_0\right) - \frac{K_I^{\tau R}}{T}\Pi_i^{\mathrm{eq}}\right)T \\ & \quad - \left(R_0 + \left(K_P^{\theta} + K_I^{\theta}\right)\left(\frac{1}{N}\sum_{k=1}^{N}\tilde{P}_k^{\mathrm{eq}} - \tilde{P}_i^{\mathrm{eq}} + K_I^{\theta}\phi_i^{\mathrm{eq}}\right)T\right) \end{cases}, \tag{5.45}$$

which leads to the following expressions at equilibrium.

$$\begin{cases} \delta\beta_i^{\mathrm{eq}} & = & 0 \\ B_i^{\mathrm{eq}} - \tau_0\tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}} & = & 0 \\ \tilde{P}_i^{\mathrm{eq}} - \frac{1}{N}\sum_{k=1}^{N}\tilde{P}_k^{\mathrm{eq}} & = & 0 \\ \tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}} - R_0 + \frac{K_I^{\tau R}}{T}\Pi_i^{\mathrm{eq}} & = & 0 \\ \tilde{P}_i^{\mathrm{eq}} - \tilde{\beta}_i^{\mathrm{eq}} + \gamma K_I^{\tau R}\Pi_i^{\mathrm{eq}} & = & 0 \\ K_I^{\tau R}\Pi_i^{\mathrm{eq}} + K_I^{\theta}T\phi_i^{\mathrm{eq}} & = & 0 \end{cases} \tag{5.46}$$

This is a system of six equations with six unknowns, since all β_i^{eq} are known source characteristics.

The system resolution in (5.46) can be rewritten as

$$\begin{pmatrix} I & 0 & 0 & 0 & 0 & 0 \\ 0 & I & -\tau_0 I & 0 & 0 & 0 \\ 0 & 0 & 0 & L & 0 & 0 \\ 0 & 0 & I & 0 & \frac{K_I^{\tau R}}{T}I & 0 \\ 0 & 0 & 0 & I & \gamma K_I^{\tau R} & 0 \\ 0 & 0 & 0 & 0 & K_I^{\tau R}I & K_I^{\theta}TI \end{pmatrix} \begin{pmatrix} \delta\beta_i^{\mathrm{eq}} \\ \tilde{B}_i^{\mathrm{eq}} \\ \tilde{R}_i^{\mathrm{e}^{\mathrm{eq}}} \\ \tilde{P}_i^{\mathrm{eq}} \\ \Pi_i^{\mathrm{eq}} \\ \phi_i^{\mathrm{eq}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ R_0 \\ \tilde{\beta}_i^{\mathrm{eq}} \\ 0 \\ 0 \end{pmatrix}. \tag{5.47}$$

The matrix

$$\begin{pmatrix} I & 0 & 0 & 0 & 0 & 0 \\ 0 & I & -\tau_0 I & 0 & 0 & 0 \\ 0 & 0 & 0 & L & 0 & 0 \\ 0 & 0 & I & 0 & \frac{K_I^{\tau R}}{T} I & 0 \\ 0 & 0 & 0 & I & \gamma K_I^{\tau R} & 0 \\ 0 & 0 & 0 & 0 & K_I^{\tau R} I & K_I^{\theta} T I \end{pmatrix}$$

is a full rang matrix when $K_I^{\tau R} \neq 0$ and $K_I^{\theta} \neq 0$, thus, solutions of the system resolution in (5.46) using PI controllers exit.

At equilibrium, from $\frac{B_i^{\text{eq}}}{\tilde{R}_i^{\text{eeq}}} = \tau_0$, equal delay constraint is satisfied. From $\tilde{P}_i^{\text{eq}} = \frac{1}{N}\sum_{k=1}^{N} \tilde{P}_k^{\text{eq}}$, the quality fairness constraint among the N programs is satisfied.

The evolution of the system state around the equilibrium is

$$\begin{cases} \Delta\tilde{\beta}_i\,(j+1) = & \Delta\beta_i\,(j) + \delta\beta(j) \\ \Delta\phi_i\,(j+1) = & \Delta\phi_i\,(j) + \frac{1}{N}\sum_{k=1}^{N}\left(\Delta\tilde{P}_k\,(j)\right) - \Delta\tilde{P}_i\,(j) \\ \Delta\Pi_i\,(j+1) = & \Delta\Pi_i\,(j) - \tau_0 V \Delta\tilde{R}_i^{e}(j) + V\Delta B_i(j) \\ \Delta\tilde{R}_i^{\text{e}}(j+1) = & \left(\alpha\left(\frac{K_P^{\tau R}}{T} + \frac{K_I^{\tau R}}{T}\right)\tau_0 V + 1 - \alpha\right)\Delta\tilde{R}_i^{e}(j) - \alpha\left(\frac{K_P^{\tau R}}{T} + \frac{K_I^{\tau R}}{T}\right) V\Delta B_i(j) - \alpha\frac{K_I^{\tau R}}{T}\Delta\Pi_i\,(j) \\ \Delta\tilde{P}_i\,(j+1) = & \Delta\beta_i\,(j) + \gamma\tau_0 V\left(K_P^{\tau R} + K_I^{\tau R}\right)\Delta\tilde{R}_i^{e}(j) - \gamma V\left(K_P^{\tau R} + K_I^{\tau R}\right)\Delta B_i(j) - \gamma K_I^{\tau R}\Delta\Pi_i\,(j) \\ \Delta B_i\,(j+1) = & \left(1 - \left(K_P^{\tau R} + K_I^{\tau R}\right)V\right)\Delta B_i(j) + \left(K_P^{\tau R} + K_I^{\tau R}\right)\tau_0 V\Delta\tilde{R}_i^{e}(j) - K_I^{\tau R}\Delta\Pi_i\,(j) \\ & - \left(\left(K_P^{\theta} + K_I^{\theta}\right)\left(\frac{1}{N}\sum_{k=1}^{N}\Delta\tilde{P}_k\,(j) + \Delta\tilde{P}_i\,(j)\right) + K_I^{\theta}\Delta\phi_i\,(j)\right) T \end{cases} \tag{5.48}$$

with $V = \frac{1}{\tilde{R}_i^{\text{eeq}}} = \frac{N}{R^{\text{c}}}$.

4.3.2 Stability

The stability analysis of the linearized control system requires its discrete-time state-space representation

$$\begin{aligned} \Delta X^{\tau}(j+1) &= A^{\tau}\Delta X^{\tau}(j) + B^{\tau}\Delta U^{\tau}(j) \\ \Delta Y^{\tau}(j) &= C^{\tau}\Delta X^{\tau}(j) \end{aligned} \tag{5.49}$$

with

$$\begin{aligned} \Delta X^{\tau}(j) = \Big(&\Delta\tilde{\beta}_1\,(j), \ldots, \Delta\tilde{\beta}_N\,(j), \Delta\Pi_1(j), \ldots \Delta\Pi_N(j), \Delta\phi_1\,(j), \ldots, \Delta\phi_N\,(j), \\ &\Delta\tilde{R}_1^{\text{e}}\,(j), \ldots, \Delta\tilde{R}_N^{\text{e}}\,(j), \tilde{P}_1\,(j), \ldots, \Delta\tilde{P}_N\,(j), \Delta B_1\,(j), \ldots, \Delta B_N\,(j)\Big)^{T}, \end{aligned} \tag{5.50}$$

and

$$\delta U^{\tau}(j) = \left(\delta\beta_1\,(j) \ldots \delta\beta_N\,(j)\right)^{T}. \tag{5.51}$$

In the considered system, we assume that the state of the buffer, the encoding rate, and the PSNR of the encoded VUs may be measured, thus,

$$\mathrm{A}^{\tau} = \begin{pmatrix} I & 0 & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & -L & 0 \\ 0 & 0 & I & -\tau_0 V I & 0 & V I \\ 0 & 0 & -\alpha\frac{K_I^{\tau R}}{T} I & \left(\alpha\left(\frac{K_P^{\tau R}}{T} + \frac{K_I^{\tau R}}{T}\right)\tau_0 V + 1 - \alpha\right) I & 0 & -\alpha\left(\frac{K_P^{\tau R}}{T} + \frac{K_I^{\tau R}}{T}\right) V I \\ I & 0 & -\gamma K_I^{\tau R} I & \gamma\tau_0 V\left(K_P^{\tau R} + K_I^{\tau R}\right) I & 0 & -\gamma V\left(K_P^{\tau R} + K_I^{\tau R}\right) I \\ 0 & -K_I^{\theta} T I & -K_I^{\tau R} I & \left(K_P^{\tau R} + K_I^{\tau R}\right)\tau_0 V I & \left(K_P^{\theta} + K_I^{\theta}\right) T L & \left(1 - \left(K_P^{\tau R} + K_I^{\tau R}\right) V\right) I \end{pmatrix}, \tag{5.52}$$

$$B^{\tau} = \begin{pmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{5.53}$$

and

$$C^{\mathrm{b}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{pmatrix} \tag{5.54}$$

After some derivations (Appendix B), one gets

$$\det\left(zI - A^{\tau}\right) = (z-1)^{2N}\left(P_5(z)\,P_2(z) - P_3(z)\,P_4(z)\right)^N \tag{5.55}$$

with

$$\begin{cases} P_2(z) = & a_1 z^4 + (a_2 - a_1) z^3 + (a_3 - a_2) z^2 - a_3 z \\ P_3(z) = & \gamma V K^{\tau R} a_1 z^3 + \gamma V \left(a_2 K^{\tau R} - K_P^{\tau R} a_1 + \frac{\tau_0 V K^{\tau R} \alpha K^{\tau R}}{T}\right) z^2 \\ & + \gamma V \left(K^{\tau R} a_3 - K_P^{\tau R} a_2 - 2\frac{\alpha K_P^{\tau R} V^2 \gamma \tau_0 K^{\tau R}}{T}\right) z + \frac{\gamma V \alpha V \tau_0 K_P^{\tau R} K_P^{\tau R}}{T} - K_P^{\tau R} a_3 \\ P_4(z) = & -K_I^{\theta} T a_1 z^3 - \left(2K_I^{\theta} T a_1 + a_2 K_I^{\theta} T\right) z^2 - \left(2K_I^{\theta} T a_2 + K_I^{\theta} T a_3\right) z - 2K_I^{\theta} T a_3 \\ P_5(z) = & a_1 z^4 + \left(a_1\left(K^{\tau R} V - 2\right) + a_2\right) z^3 + \left(a_1\left(1 - K_P^{\tau R} V\right) + a_2\left(K^{\tau R} V - 2\right) + a^3 + \frac{\tau_0 V^2 \alpha}{T} K^{\tau R} K^{\tau R}\right) z^2 \\ & + \left(\left(1 - K_P^{\tau R} V\right) a_2 + \left(K^{\tau R} V - 2\right) a_3 - 2\frac{\tau_0 V^2 \alpha}{T} K^{\tau R} K_P^{\tau R}\right) z + 1 - K_P^{\tau R} V a_3 + K_P^{\tau R} K_P^{\tau R} \frac{\tau_0 V^2 \alpha}{T} \end{cases}, \tag{5.56}$$

and

$$\begin{cases} a_1 = & 1 \\ a_2 = & -\alpha \frac{K^{\tau R}}{T} \tau_0 V - 2 + \alpha \\ a_3 = & \alpha \frac{K^{\tau R}}{T} \tau_0 V + 1 - \alpha - \alpha \tau_0 V \frac{K_I^{\tau R}}{T} \end{cases} \tag{5.57}$$

We assume that at equilibrium $\frac{1}{V} = \tilde{R}_i^{\mathrm{eq}} = R_0$

The root $z = 1$ is located on the unit circle and it is related to the dynamic of the source. Roots of the second part in (5.55) can be obtained numerically. Using these roots, the set of control parameters K_P^{θ}, K_I^{θ}, $K_P^{\tau R}$, and $K_I^{\tau R}$ ensuring that the system is stable can be determined.

5 Control at the frame level

When the control is performed at the frame level, the bandwidth allocation control proposed in (5.2) should be performed while considering the heterogeneity of the video sequence. In fact, I frames and P frames may have different impact on the PSNR and the encoding rate values. In addition, I frames may not be synchronized among the multiplexed programs. Using bandwidth allocation in (5.2) may results in long delay between the encoding time and the decoding time for I frame. Consider all multiplexed programs are segmented into GoPs with N_G is the GoP size. Consider an example of GoP structure (IPPPP...IPPPP...) with only I and P frames.

To tackle the heterogeneity issue and get the transmission rate for each frame, we use the PSNR history over a sliding window including the N_G previous frames so that

$$\mathbf{P}_i(j-1) = \begin{pmatrix} P_i(j-1) \\ \vdots \\ P_i(j-N_G) \end{pmatrix} \tag{5.58}$$

and

$$\bar{\mathbf{P}}_i(j-1) = \begin{pmatrix} \bar{P}(j-1) \\ \vdots \\ \bar{P}(j-N_G) \end{pmatrix} = \frac{1}{N} \begin{pmatrix} \sum_{k=1}^{N} P_k(j-1) \\ \vdots \\ \sum_{k=1}^{N} P_k(j-N_G) \end{pmatrix}. \tag{5.59}$$

Using a PI controller, the allocated rates for the j-th frame are

$$R_i^{\mathrm{t}}(j) = R_0 + K_{\mathrm{P}}^{\theta} \frac{1}{N_G} \sum_{l=1}^{N_G} (\bar{P}(j-l) - P_i(j-l)) + K_{\mathrm{I}}^{\theta} \frac{1}{N_G} \sum_{l=j-N_G}^{j-1} \sum_{k=1}^{l} (\bar{P}(k) - P_i(k)), \tag{5.60}$$

Notice that, the control at the frame level requires R-D models at the frame level. These models should take into account the impact of the chosen encoding parameter in the reference frame on the rate and the distortion of its corresponding predicted frame at the expense of an increased modeling complexity than with a GoP-level control.

Using encoder control in (5.3) may results in some differences in the video quality resulting from the chosen encoding rate.

The encoding rate should also be performed over a sliding window including the N_G previous frames so that

$$\mathbf{B}_i(j) = \begin{pmatrix} B_i(j) \\ \vdots \\ B_i(j-N_G+1) \end{pmatrix} \tag{5.61}$$

and

$$\bar{\mathbf{B}}_i(j) = \begin{pmatrix} \bar{B}(j) \\ \vdots \\ \bar{B}(j-N_G+1) \end{pmatrix} = \frac{1}{N} \begin{pmatrix} \sum_{k=1}^{N} B_k(j) \\ \vdots \\ \sum_{k=1}^{N} B_k(j-N_G+1) \end{pmatrix}. \tag{5.62}$$

Using a PI controller, the encoding rates for the j-th frame are

$$R_i^{\mathrm{e}}(j) = \frac{R^c}{N} - K_{\mathrm{P}}^{\mathrm{bR}} \frac{1}{N_G} \sum_{l=0}^{N_G-1} \frac{B_i(j-l)-B_0}{T} - K_{\mathrm{I}}^{\mathrm{bR}} \frac{1}{N_G} \sum_{l=j-N_G+1}^{j} \sum_{k=1}^{l} \left(\frac{B_i(k)-B_0}{T} \right), \tag{5.63}$$

Here as in the control of the buffer level in bits, the buffer level in seconds can be controlled at the frame level using the sliding window including the N_G previous frames.

6 Experimental results

In this section, we evaluate the performance of the proposed distributed SM system using first, Gaussian sources, and then, video sources while considering both P and PI controllers and buffer level control in bits and in seconds.

6.1 Gaussian sources

The performance of the proposed SM system is evaluated first with $N = 3$ Gaussian sources each one being characterized by a time-varying variance to simulate variations of the video sources. Small variations are modeled by adding variances $z_i(j)$ uniformly distributed in the interval $[-1, 1]$, so that

$$\sigma_i^{z\,2}(j) = \sigma_i^2(j) + z_i(j). \tag{5.64}$$

From time to time, a jump in the variance is performed in order to simulate abrupt scene changes in the video program, see Figure 5.4. In this part $T = 1$ s, $\gamma = 6.02$ dB/bits, and channel rate $R^c = 10$ bits s^{-1}. Here a VU is a single source sample.

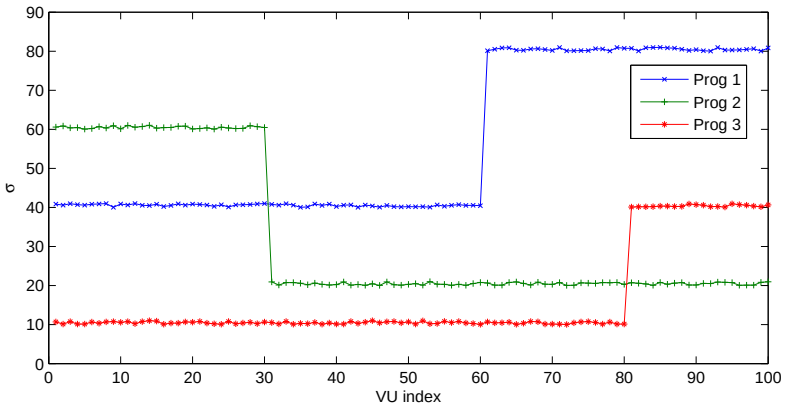


Figure 5.4: The instantaneous variance σ^2 of the $N = 3$ Gaussian sources

6.1.1 Buffer levels in bits

We first focus on the system performance using buffer levels in bits to update the encoding rate. Parameters of the bandwidth allocation and the encoder control using P controllers and those using PI controllers are set in such a way that the corresponding roots, solutions of (5.35) using $K_{\mathrm{P}}^{\theta} = 0.7$, $K_{\mathrm{I}}^{\theta} = 0.1$, $K_{\mathrm{P}}^{\mathrm{bR}} = 0.2$, and $K_{\mathrm{I}}^{\mathrm{bR}} = 0.08$, are located in the unit circle, see Figure 5.9.

The transmission rate allocation is performed according to (5.2) using a PI controller with $K_{\mathrm{P}}^{\theta} = 0.7$ and $K_{\mathrm{I}}^{\theta} = 0.1$ corresponding to the least average variance of the PSNR over the N programs denoted by σ_{PSNR}^2 and expressed by

$$\sigma_{PSNR}^2 = \frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{M}\sum_{l=1}^{M}\left(P_{nl} - \bar{P}_l\right)^2\right), \tag{5.65}$$

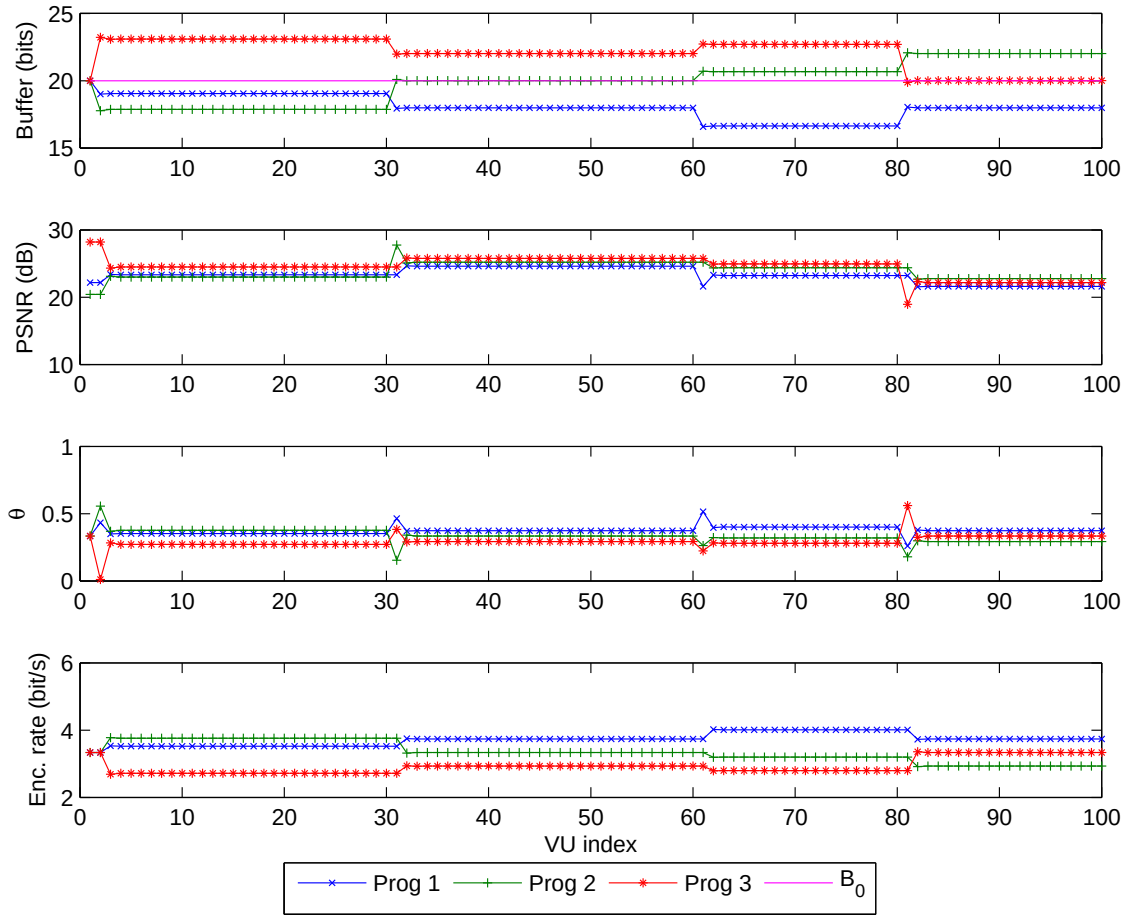


Figure 5.5: Behavior of the proposed system using P controllers for both bandwidth allocation and encoding rate (P_P), when the encoding rate is adjusted according to the buffer levels in bits using $K_{\mathrm{P}}^{\theta} = 0.7$ and $K_{\mathrm{P}}^{\mathrm{bR}} = 0.2$.

and low PSNR discrepancy denoted by Δ_{PSNR}

$$\Delta_{PSNR} = \left| \frac{1}{N} \sum_{n=1}^{N} \left(\frac{1}{M} \sum_{l=1}^{M} \left(P_{nl} - \bar{P}_l \right) \right) \right|, \tag{5.66}$$

where M is the number of VUs per sequence and $\bar{P}_n$ is the average PSNR over the M VUs in Program n. The encoding rates are obtained from (5.3) using PI controller with $K_{\mathrm{P}}^{\mathrm{bR}} = 0.2$ and $K_{\mathrm{I}}^{\mathrm{bR}} = 0.08$. This choice of K_{P}^{θ}, K_{I}^{θ}, $K_{\mathrm{P}}^{\mathrm{bR}}$, and $K_{\mathrm{I}}^{\mathrm{bR}}$ leads to roots in the unit circle see Figure 5.9.

When considering $K_{\mathrm{P}}^{\theta} = 3$, $K_{\mathrm{P}}^{\mathrm{bR}} = 0.5$, located in the instability region of the P control system obtained in Section 4.2, the system behavior is reported in Figure 5.8. Using these parameter values the system becomes oscillating.

Among the values of the parameters leading to a stable system, parameters of the P and the PI controllers may be tunned manually to reach good performance in terms discrepancy of the buffer levels in bits, variance of the buffer levels in bits, PSNR discrepancy, and PSNR variance. Buffers have been assumed to be initially at their reference level B_0. The parameter $K_{\mathrm{P}}^{\mathrm{bR}}$ is tuned first to maximize the rise speed to some equilibrium while having no overshoot. Then, the parameter $K_{\mathrm{I}}^{\mathrm{bR}}$ is

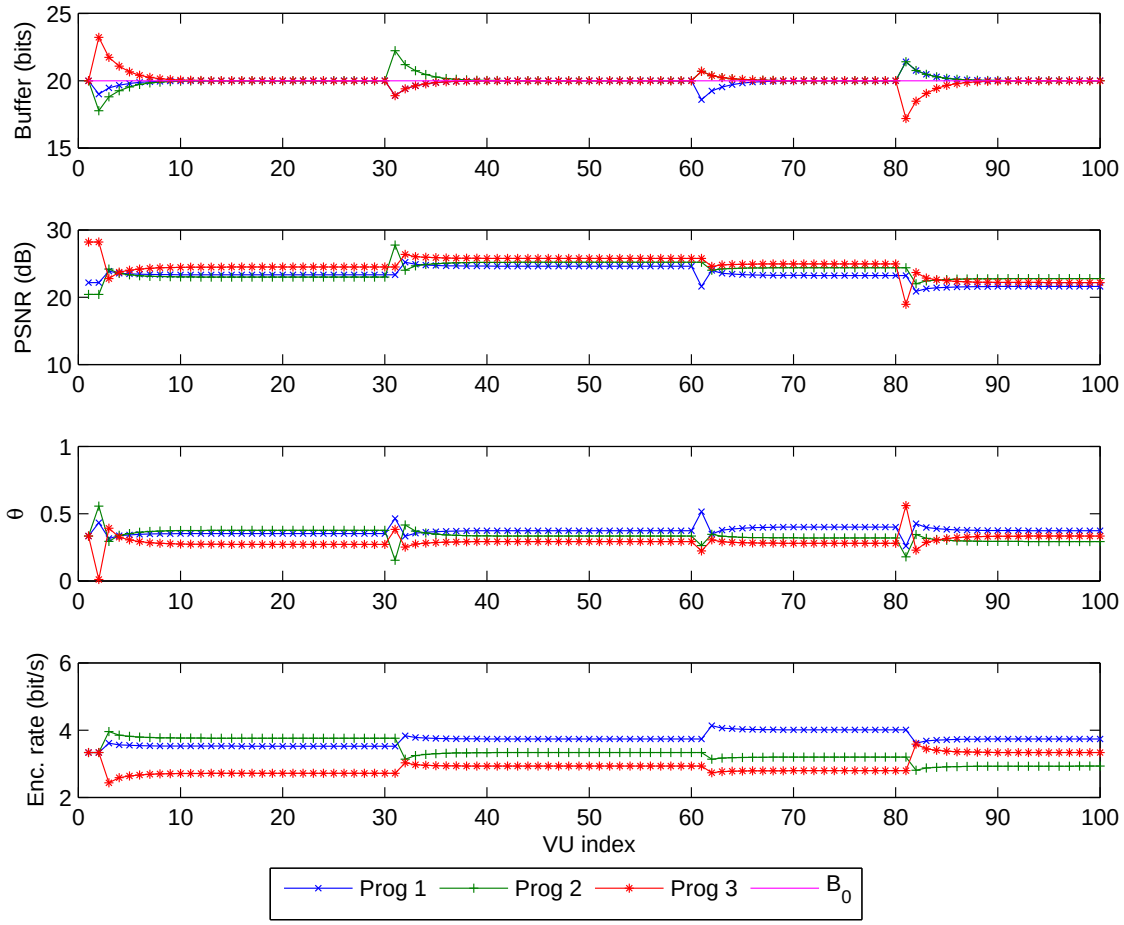


Figure 5.6: Behavior of the proposed system using a P controller for bandwidth allocation and a PI controller for encoding rate (P_PI), when the encoding rate is adjusted according to the buffer levels in bits using $K_{\mathrm{P}}^{\theta} = 0.7$, $K_{\mathrm{P}}^{\mathrm{bR}} = 0.2$, and $K_{\mathrm{I}}^{\mathrm{R}} = 0.08$.

adjusted to eliminate the offset and to minimize the overshoot.

The buffer levels in bits, the PSNR, the rate allocation, and the encoding rate of each program when the buffer bit level is controlled are represented in

- Figure 5.5, when P controllers are used in both bandwidth allocation and encoder control (P_P),
- Figure 5.6, when a P controller is used in the bandwidth allocation control and a PI controller in the encoder control (P_PI),
- Figure 5.7 when PI controllers are used in both bandwidth allocation and encoder control (PI_PI).

In these curves, the reference buffer level in bits is set to $B_0 = 20$ bits.

From Figure 5.5, we notice that the control system reaches equilibrium after short transients to satisfy the quality fairness constraint.

Using a P controller in the feedback control system of the rate control, a significant discrepancy between the bit level of the buffers at equilibrium and the reference buffer bit level occurs. This discrepancy is fully compensated as shown in Figure 5.6

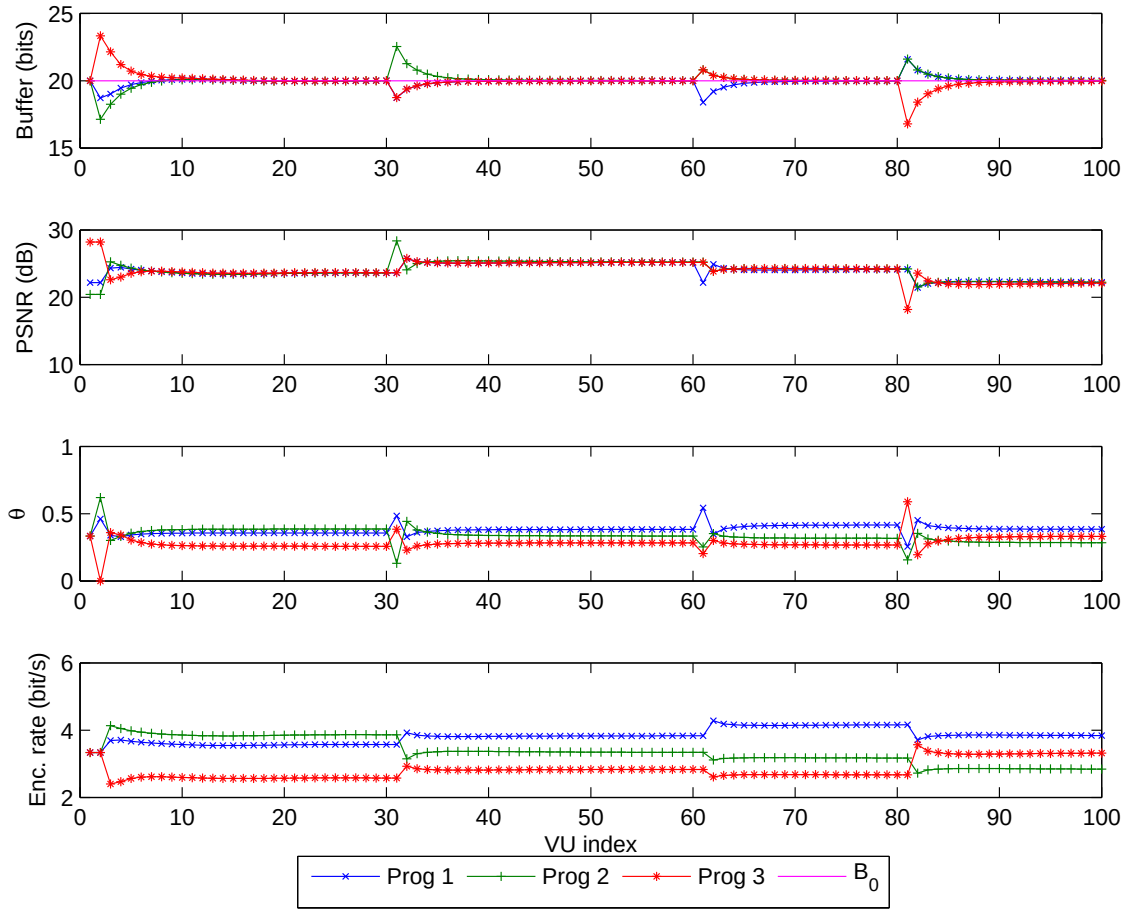


Figure 5.7: Behavior of the proposed system using PI controllers for both bandwidth allocation and encoding rate (PI_PI), when the encoding rate is adjusted according to the buffer levels in bits using $K_{\mathrm{P}}^{\theta} = 0.7$, $K_{\mathrm{I}}^{\theta} = 0.1$, $K_{\mathrm{P}}^{\mathrm{bR}} = 0.2$, and $K_{\mathrm{I}}^{\mathrm{bR}} = 0.08$.

by using a PI control instead of a P control in the encoder control process. Then, when a PI is used in the bandwidth allocation in Figure 5.7, at equilibrium, almost equal PSNR are obtained for all programs. The system performance is measured in terms of the absolute value of the averaged discrepancy of the buffer levels in bits among the N programs denoted by Δ_B, the averaged variance of the buffer levels in bits among the N programs denoted by σ_B^2, the averaged PSNR variance among the N programs denoted by σ_{PSNR}^2, and the averaged PSNR discrepancy among the N programs denoted by Δ_{PSNR}. With

$$\Delta_B = \left| \frac{1}{N} \sum_{n=1}^{N} \left(\frac{1}{M} \sum_{l=1}^{M} (B_{nl} - B_0) \right) \right|, \tag{5.67}$$

$$\sigma_B^2 = \frac{1}{N} \sum_{n=1}^{N} \left(\frac{1}{M} \sum_{l=1}^{M} (B_{nl} - B_0)^2 \right). \tag{5.68}$$

The system performance are summarized in Table 5.1, when using (P_P), (P_PI), and (PI_PI) controllers.

From Table 5.1, one can see that, using a PI controller in the encoder control, the buffer level discrepancy and the buffer level variance decrease significantly compared

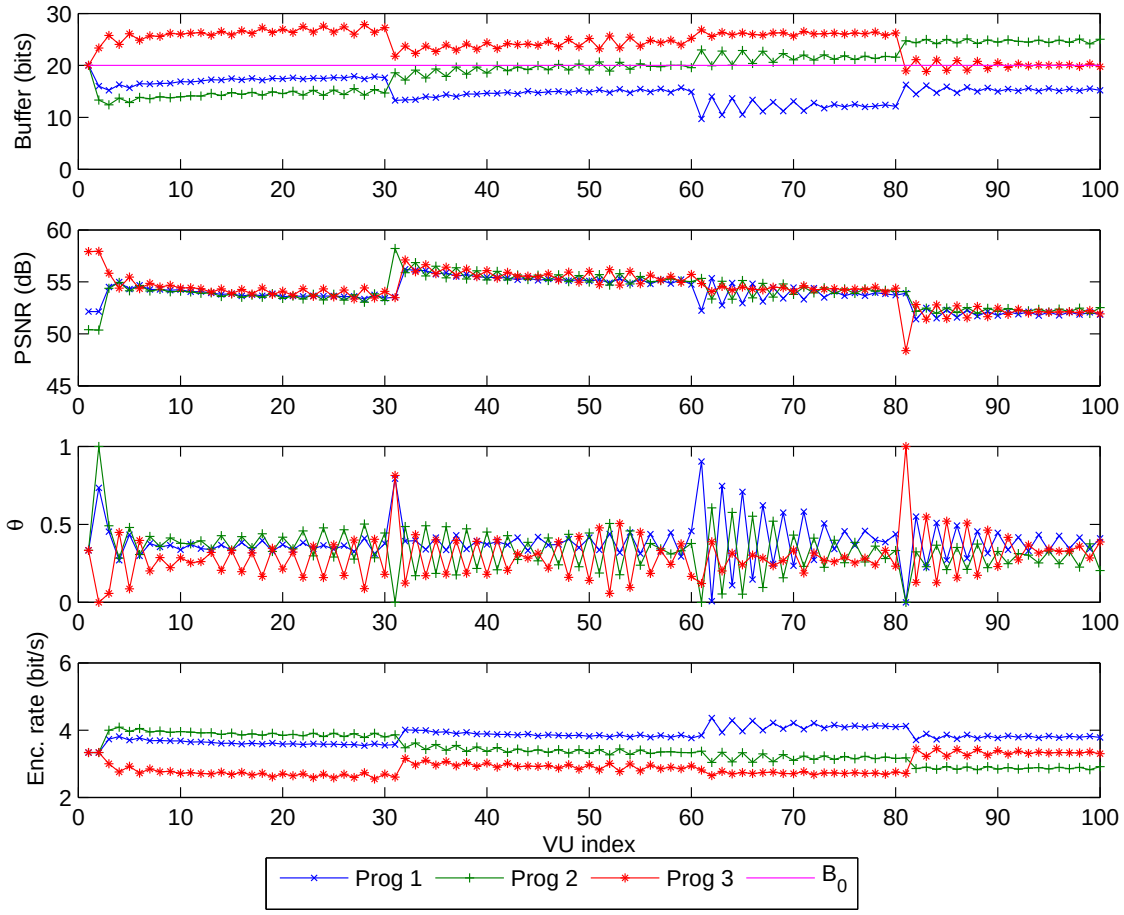


Figure 5.8: Behavior of the proposed system using P controllers for both bandwidth allocation and encoding rate (P_P), when the encoding rate is adjusted according to the buffer levels in bits using $K_{\mathrm{P}}^{\theta} = 3$, $K_{\mathrm{I}}^{\theta} = 0$, $K_{\mathrm{P}}^{\mathrm{bR}} = 0.5$, and $K_{\mathrm{I}}^{\mathrm{bR}} = 0$.

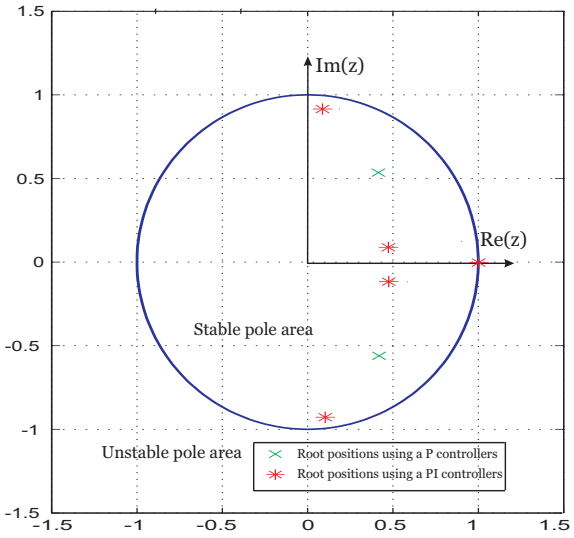


Figure 5.9: Position of the roots using a (P_P) and (PI_PI) controller using Gaussian sources, when buffer levels in bits are controlled ($K_{\mathrm{P}}^{\theta} = 0.7$, $K_{\mathrm{I}}^{\theta} = 0.1$, $K_{\mathrm{P}}^{\mathrm{bR}} = 0.2$, and $K_{\mathrm{I}}^{\mathrm{bR}} = 0.08$).

$K_{\mathrm{P}}^{\theta}, K_{\mathrm{I}}^{\theta}$	$K_{\mathrm{P}}^{\mathrm{bR}}, K_{\mathrm{I}}^{\mathrm{bR}}$	Δ_B	σ_B^2	Δ_{PSNR}	σ_{PSNR}^2
$0.7, 0$	$0.2, 0$	1.96	1.42	0.59	1.56
$0.7, 0$	$0.2, 0.08$	0.05	0.18	0.57	1.57
$0.7, 0.1$	$0.2, 0.08$	0.06	0.24	0.05	1.56

Table 5.1: System performance in terms of Δ_B, σ_B^2, Δ_{PSNR}, and σ_{PSNR}^2 when using (P_P), (P_PI), and (PI_PI) controllers, when the encoding rate is adjusted according to the buffer levels in bits using Gaussian sources.

to when using a P controller while the PSNR discrepancy and the PSNR variance values are almost unchanged. Using a PI controller in the bandwidth allocation control, the PSNR discrepancy decreases significantly compared to when using a P controller at the expense of some buffer level variance increase. The PSNR variance value is kept unchanged.

6.1.2 Buffer levels in seconds

In this part, we focus on the system performance when considering the buffer levels in seconds to update the encoding rate.

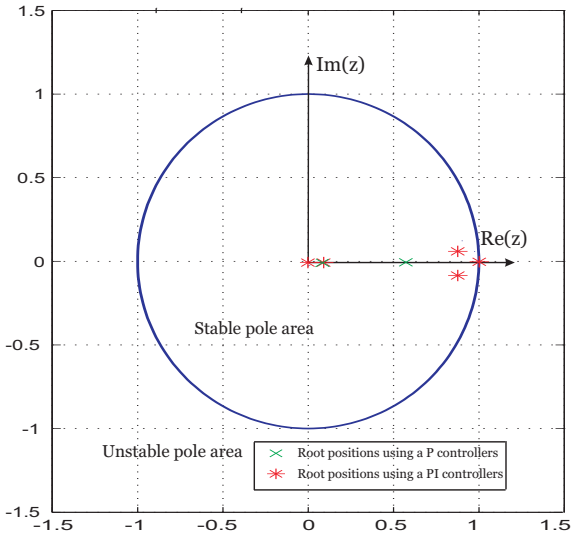


Figure 5.10: Position of the roots using a (P_P) and (PI_PI) controller using Gaussian sources, when buffer levels in seconds are controlled ($K_{\mathrm{P}}^{\theta} = 0.5$, $K_{\mathrm{I}}^{\theta} = 0.03$, $K_{\mathrm{P}}^{\tau\mathrm{R}} = 0.15$, and $K_{\mathrm{I}}^{\tau\mathrm{R}} = 0.02$).

Parameters of the bandwidth allocation and the encoder control using P controllers and those using PI controllers are set in such a way that the corresponding roots, solutions of (5.55) using $K_{\mathrm{P}}^{\theta} = 0.5$, $K_{\mathrm{I}}^{\theta} = 0.03$, $K_{\mathrm{P}}^{\tau\mathrm{R}} = 0.15$, and $K_{\mathrm{I}}^{\tau\mathrm{R}} = 0.02$, are located in the unit circle, see Figure 5.10.

The transmission rate allocation is performed using a PI controller with $K_{\mathrm{P}}^{\theta} = 0.5$ and $K_{\mathrm{I}}^{\theta} = 0.03$. The encoding rates are controlled using a PI controller with $K_{\mathrm{P}}^{\tau\mathrm{R}} = 0.15$ and $K_{\mathrm{I}}^{\tau\mathrm{R}} = 0.02$.

The buffer levels in bits, the PSNR, the rate allocation θ, the encoding rate, and the buffer levels in seconds of each program are represented in

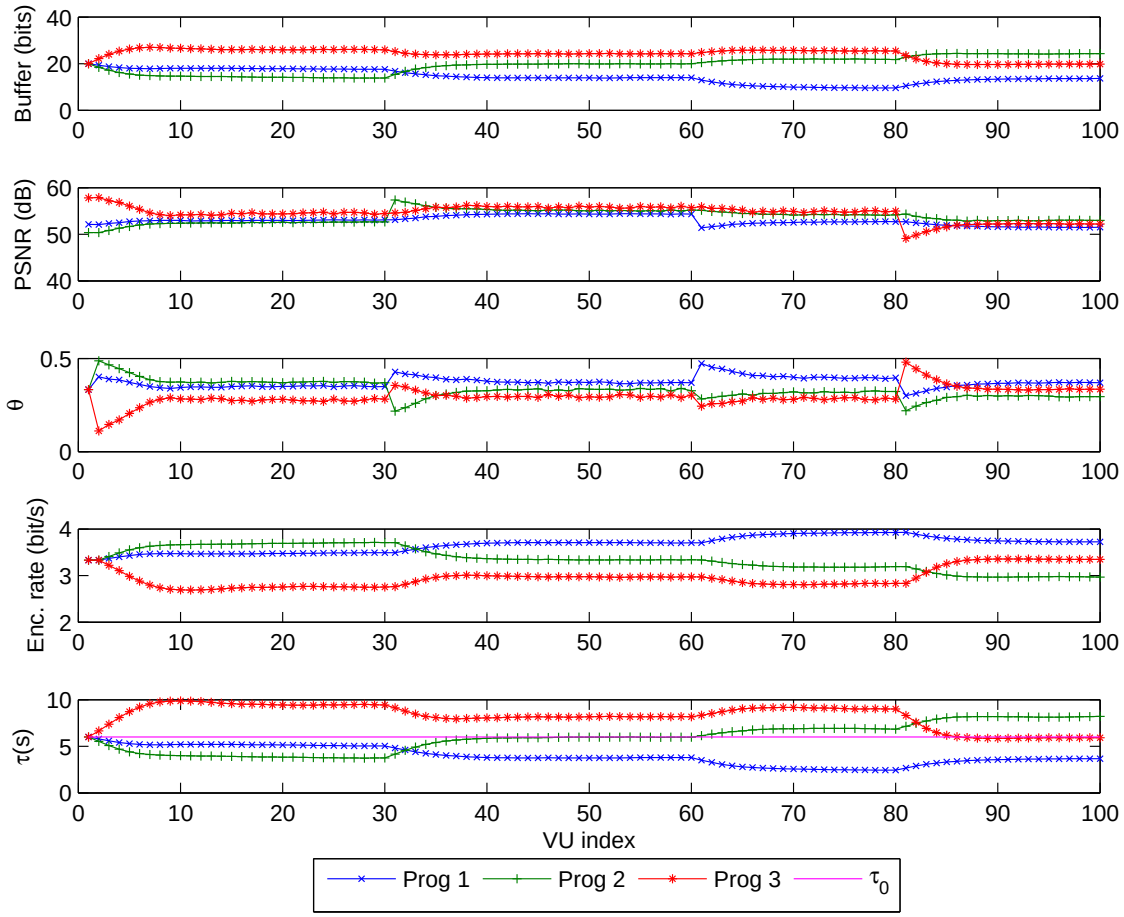


Figure 5.11: Behavior of the proposed system using P controllers for both bandwidth allocation and encoding rate controls (P_P), when the encoding rate is adjusted according to the buffer levels in seconds using $K_{\mathrm{P}}^{\theta} = 0.5$ and $K_{\mathrm{P}}^{\tau\mathrm{R}} = 0.15$.

- Figure 5.11, when P controllers are used in both bandwidth allocation and encoder control (P_P),
- Figure 5.12, when a P controller is used in the bandwidth allocation control and a PI controller in the encoder control (P_PI),
- Figure 5.13, when PI controllers are used in both bandwidth allocation and encoder control (PI_PI).

The reference delay is set to $\tau_0 = \frac{B_0 N}{R^{\mathrm{c}}} = 6T = 6$ s.

As in the previous control case, the control system reaches equilibrium after short transient modes to satisfy the quality fairness constraint. Using a P controller in the feedback control system of the rate control, a significant discrepancy between the buffer levels in seconds at equilibrium and the reference level τ_0 occurs as well as between the PSNR of the three multiplexed programs.

The absolute value of the average discrepancy of the buffer levels in seconds among the N programs denoted by Δ_τ

$$\Delta_\tau = \left| \frac{1}{N} \sum_{n=1}^{N} \left(\frac{1}{M} \sum_{l=1}^{M} (\tau_{nl} - \tau_0) \right) \right|, \tag{5.69}$$

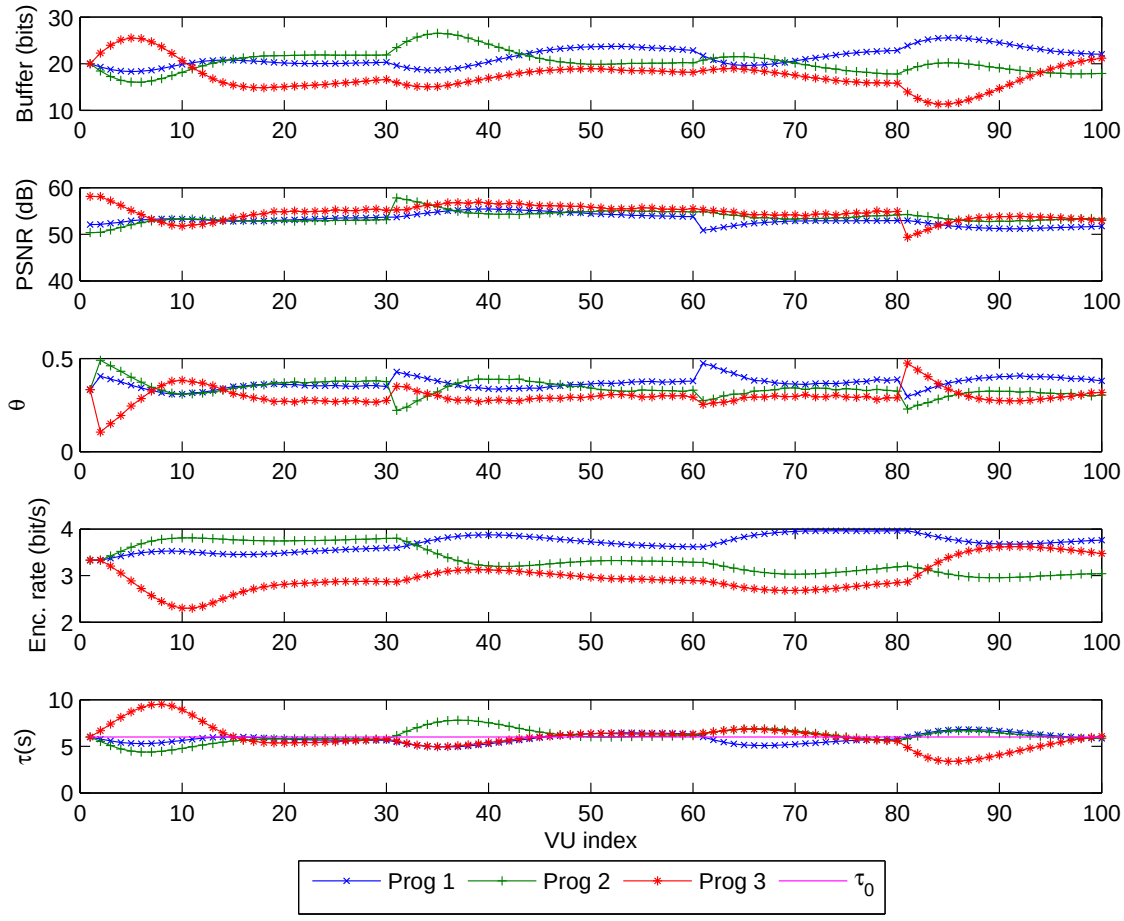


Figure 5.12: Behavior of the proposed system using P controller for both bandwidth allocation and a PI controller for encoding rate control (P_PI), when the encoding rate is adjusted according to the buffer levels in seconds using $K_{\mathrm{P}}^{\theta} = 0.5$, $K_{\mathrm{P}}^{\tau\mathrm{R}} = 0.15$, and $K_{\mathrm{I}}^{\tau\mathrm{R}} = 0.02$.

are fully compensated as shown in Figure 5.12 by using a PI control instead of a P control in the encoder control process. The system performance is measured in terms of the average discrepancy of the buffer levels in seconds denoted by Δ_τ, the averaged variance of the buffer levels in seconds among the N programs denoted by σ_τ^2, the average PSNR variance denoted by σ_{PSNR}^2, and the average PSNR discrepancy denoted by Δ_{PSNR}. With

$$\sigma_\tau^2 = \frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{M}\sum_{l=1}^{M}\left(\tau_{nl}-\tau_0\right)^2\right), \tag{5.70}$$

System performance is summarized in Table 5.2, when using (P_P), (P_PI), and (PI_PI) controllers.

From Table 5.2, one can see that, using a PI controller in the encoder control, the buffer level discrepancy and the buffer level variance decrease significantly compared to when using a P controller while the PSNR discrepancy and the PSNR variance values are almost unchanged. Using a PI controller in the bandwidth allocation control, the PSNR discrepancy decreases significantly compared to when using a P controller at the expense of some buffer level discrepancy and variance increase.

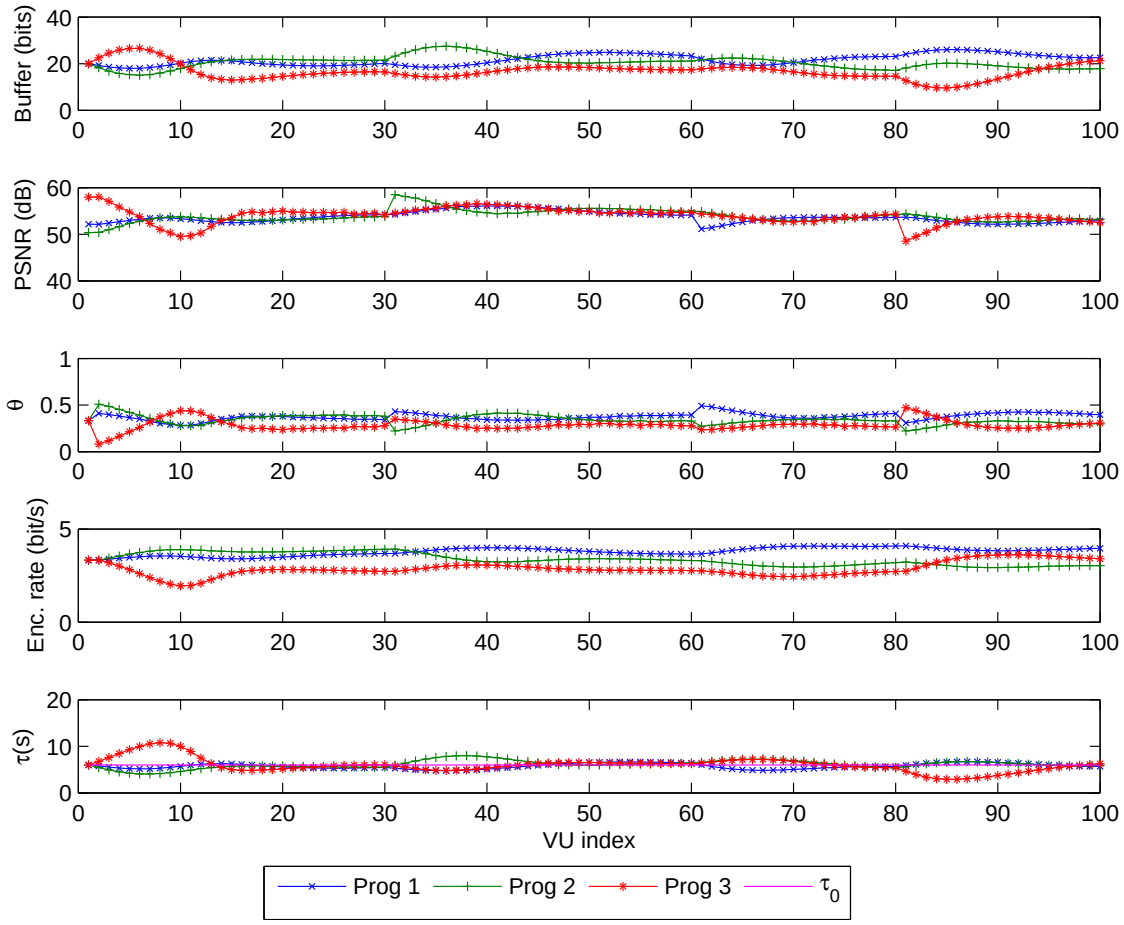


Figure 5.13: Behavior of the proposed system using PI controllers for both bandwidth allocation and encoding rate controls (PI_PI), when the encoding rate is adjusted according to the buffer levels in seconds using $K_{\mathrm{P}}^{\theta} = 0.5$, $K_{\mathrm{I}}^{\theta} = 0.03$, $K_{\mathrm{P}}^{\tau\mathrm{R}} = 0.15$, and $K_{\mathrm{I}}^{\tau\mathrm{R}} = 0.02$.

$K_{\mathrm{P}}^{\theta}, K_{\mathrm{I}}^{\theta}$	$K_{\mathrm{P}}^{\tau\mathrm{R}}, K_{\mathrm{I}}^{\tau\mathrm{R}}$	Δ_{τ}	σ_{τ}^2	Δ_{PSNR}	σ_{PSNR}^2
$0.5, 0$	$0.15, 0$	2.037	1.58	0.81	1.85
$0.5, 0$	$0.15, 0.02$	0.17	0.77	0.7	1.77
$0.5, 0.03$	$0.15, 0.02$	0.25	1.1	0.19	1.8

Table 5.2: System performance in terms of Δ_{τ}, σ_{τ}^2, Δ_{PSNR}, and σ_{PSNR}^2, when using (P_P), (P_PI), and (PI_PI) controllers, when the encoding rate is adjusted according to the buffer levels in seconds using Gaussian sources.

6.2 Video sources

In this section, we consider an application of the proposed SM system with actual video sources. $N = 4$ programs are considered. Each program displays various video sequences in CIF format (Soccer, Container, coastguard and Hall) encoded with the H.264/AVC encoder in baseline profile at the same frame rate $F = 30$ frames/s to simulate abrupt scene changes in the video program as shown in Figure 5.14. Here, the VU is considered as a GoP with a duration $T = 0.5$ s. We consider a constant channel rate $R^{\mathrm{c}} = 1$ Mbit/s.

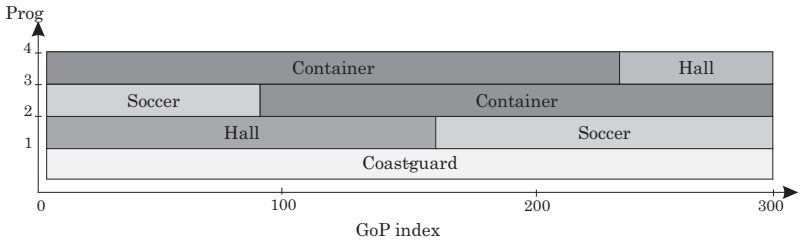


Figure 5.14: Videos transmitted over the four considered programs.

The size $B_{\max}$ of the buffers is taken large enough to support the bit level variations, occurring, *e.g.*, during scene changes. Here, their size in bits is $B_{\max} = 1$ Mbits. The reference buffer level in bits is $B_0 = 400$ kbits and the reference delay is $\tau_0 = 1$ s. Buffers are assumed initially empty. The encoding rates are initially considered equal to the reference rate $R_0 = R^c/N$. These rates correspond to the output value of the rate control process provided to each video server. The encoder is then in charge of adjusting its encoding parameters to achieve the target bit rate.

The heart of the rate control process is a model describing the relation between quantization parameter (QP) and the actual bit rate. In our simulations, we use an independent exponential R-D models, see Section 3.1 in Chapter 2, where the rate and the PSNR are expressed as functions of the QP

$$R(Q) = a_{\mathrm{R}} \exp(b_{\mathrm{R}} Q) \tag{5.71}$$

and

$$P(Q) = a_{\mathrm{P}} Q + b_{\mathrm{P}}, \tag{5.72}$$

with $(a_{\mathrm{R}}, b_{\mathrm{R}}, a_{\mathrm{P}}, b_{\mathrm{P}})$ evaluated in advance with two encoding trials for each GoP. The encoded GoP is then transmitted and stored in its corresponding buffer in the MANE. We assume that the PSNR information associated to the encoded GoP is also transmitted and available at the bandwidth allocation control process.

6.2.1 Buffer levels in bits

In the first set of experiments, we focus on the system performance when the encoding rate is adjusted according to the bit level of the buffer.

Parameters of the bandwidth allocation and the encoder control using P controllers and those using PI controllers are set in such a way that the corresponding roots, solutions of (5.35) using $K_{\mathrm{P}}^{\theta} = 3.10^4$, $K_{\mathrm{I}}^{\theta} = 4.10^3$ $K_{\mathrm{P}}^{\mathrm{bR}} = 0.6$, and $K_{\mathrm{I}}^{\mathrm{bR}} = 0.2$, are located in the unit circle, see Figure 5.17. The roots of $f(z) = \det\left(A^{\mathrm{b}} - zI\right)$ are obtained using $\gamma = 5.10^{-5}$ in (5.10) corresponding to the average PSNR-Rate model parameter evaluated in advance with two encoding trials for each GoP using Soccer sequence.

Among the values of the parameters leading to a stable system, we choose the values that provides good performance in terms of discrepancy of the buffer levels in bits, variance of the buffer levels in bits, PSNR discrepancy, and PSNR variance. K_{P}^{θ} is set first so that the least average variance of the PSNR over the N programs denoted by σ^2_{PSNR} is obtained. The integral terms allow reducing the PSNR discrepancy denoted by Δ_{PSNR}. The encoding rates are obtained from (5.3) using PI controller with $K_{\mathrm{P}}^{\mathrm{bR}} = 0.6$ and $K_{\mathrm{I}}^{\mathrm{bR}} = 0.2$. These parameters have been tuned

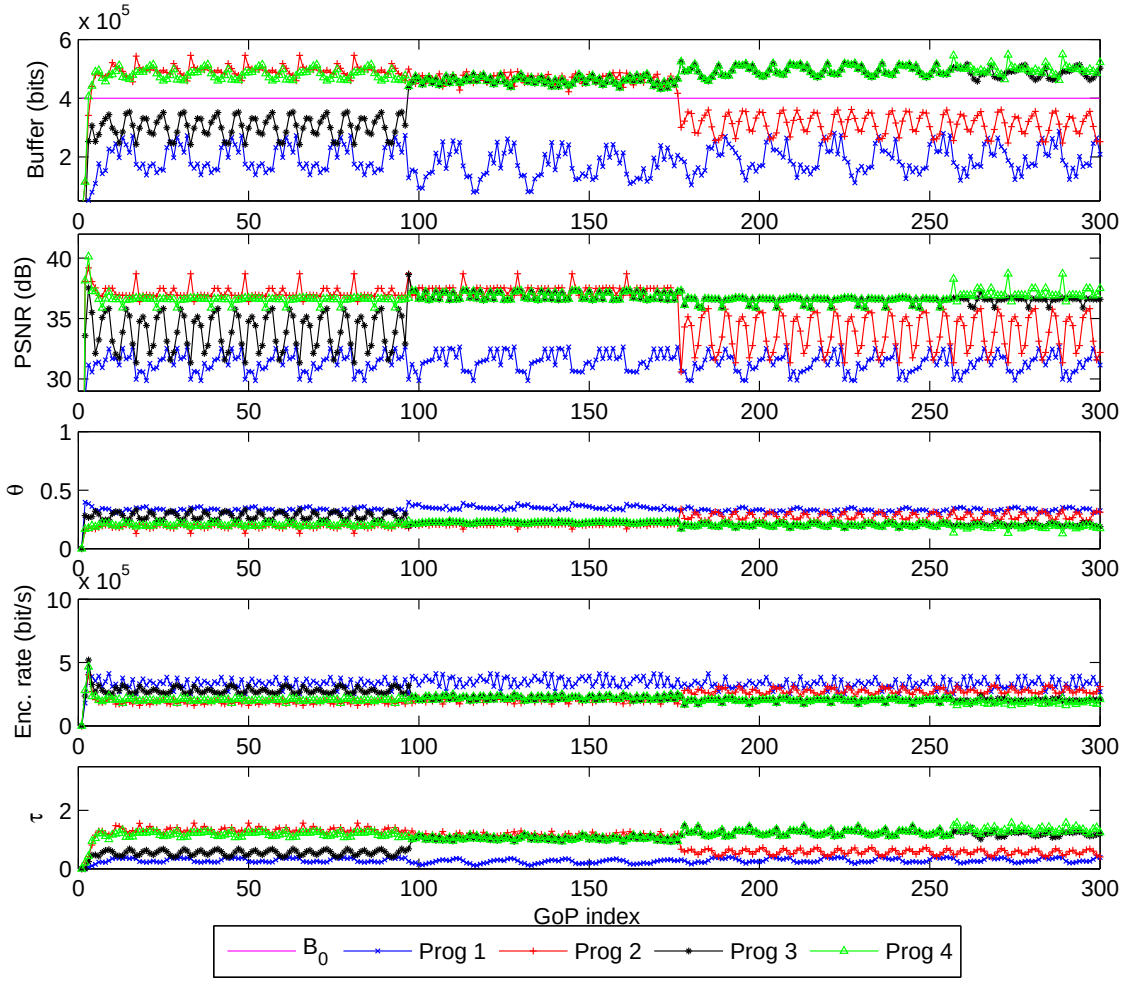


Figure 5.15: System performance using P controllers for both bandwidth allocation and encoding rate controls (P_P), when the encoding rate is adjusted according to the buffer levels in bits while statistically multiplexing four video programs using $K_{\mathrm{P}}^{\theta} = 3.10^4$, and $K_{\mathrm{P}}^{\mathrm{bR}} = 0.6$.

manually. In fact, K_{P}^{θ} and K_{I}^{θ} are tuned first to minimize the discrepancy between the PSNR of the N programs and the average PSNR. Then, the parameter K_{P}^{bR} is tunned to maximize the rise speed to the buffer level equilibrium while having no overshoot. The parameter K_{I}^{bR} is adjusted to eliminate the offset and to minimize the overshoot.

The buffer levels in bits, the PSNR, the bandwidth allocation θ, the encoding rate, and the buffer levels in seconds of each program are represented in

- Figure 5.15, when P controllers are used in both bandwidth allocation and encoder control (P_P),
- Figure 5.16, when PI controllers are used in both bandwidth allocation and encoder control (PI_PI).

The PI controller allows low discrepancy between the buffer levels in bits at equilibrium and the reference buffer level in bits B_0 for most of the buffer levels of

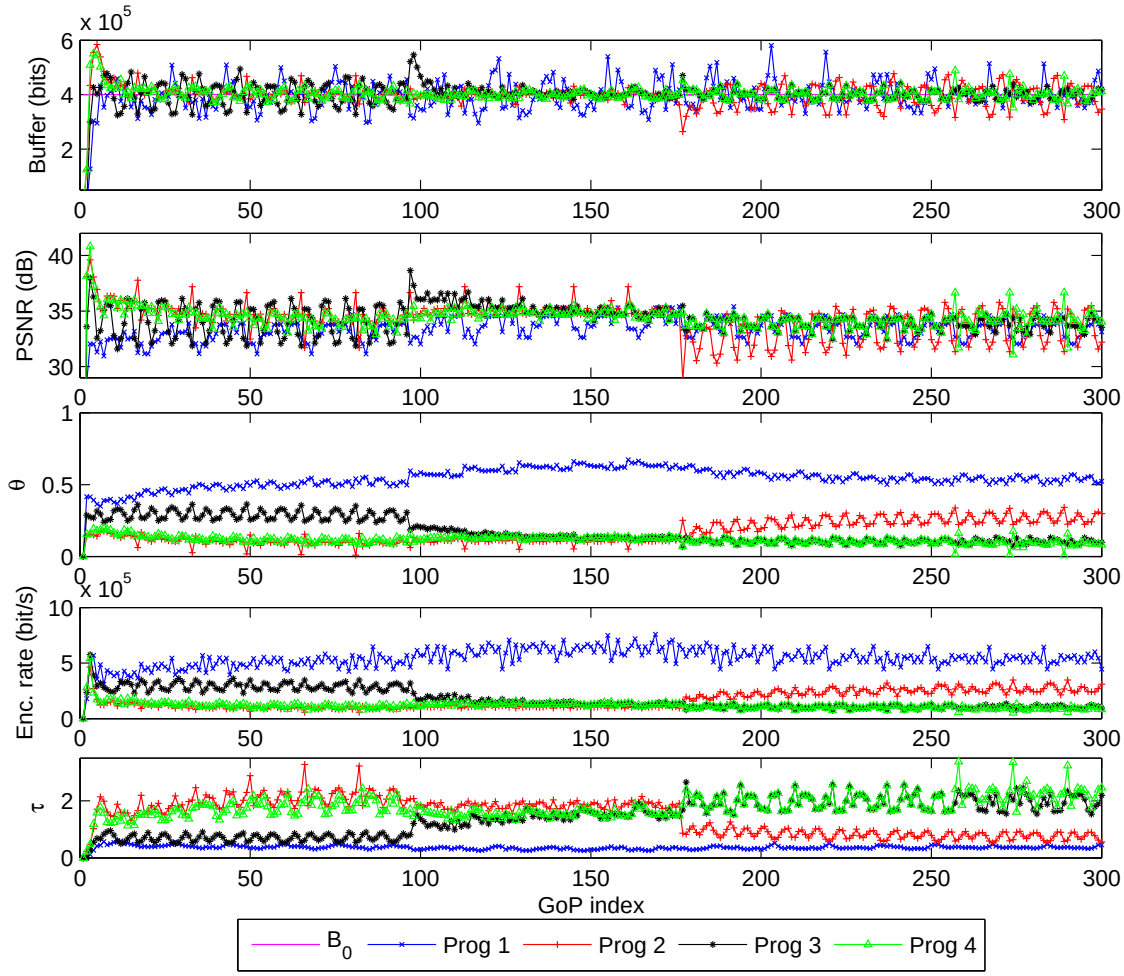


Figure 5.16: System performance using PI controllers for both bandwidth allocation and encoding rate controls (PI_PI), when the encoding rate is adjusted according to the buffer levels in bits while statistically multiplexing four video programs using $K_{\mathrm{P}}^{\theta} = 3.10^4$, $K_{\mathrm{I}}^{\theta} = 4.10^3$ $K_{\mathrm{P}}^{\mathrm{bR}} = 0.6$, and $K_{\mathrm{I}}^{\mathrm{bR}} = 0.2$.

the four considered video programs. In presence of scene change, the control system goes through some transient, see, *e.g.*, buffer bit level evolution for program 3 at GoP 97 or of program 2 at GoP 177 in Figure 5.16. After a short transient, the system gets back to an equilibrium state to allow fair quality among programs.

The system performance is measured in terms of discrepancy of the buffer levels in bits Δ_B, variance of the buffer levels in bits σ_B^2, PSNR discrepancy Δ_{PSNR} and the PSNR variance σ_{PSNR}^2. Performance is summarized in Table 5.3, when using (P_P), (P_PI), and (PI_PI) controllers. The PI controller in the encoder control process significantly decreases the discrepancy of the buffer levels in bits Δ_B compared to the P controller without disturbing the behavior of the PSNR control. In addition, the PI controller of the bandwidth allocation the PSNR discrepancy between the four multiplexed programs is significantly reduced at the expense of some buffer level discrepancy and variance increase.

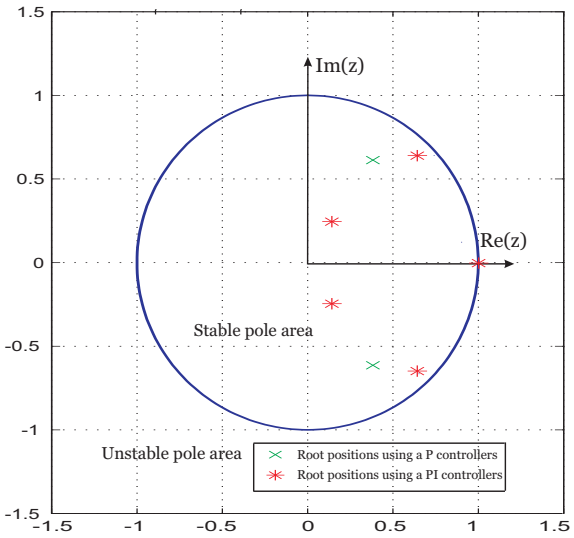


Figure 5.17: Position of the roots using (P_P) and (PI_PI) controller using video sources, when buffer levels in bits are controlled ($K_{\mathrm{P}}^{\theta} = 3.10^4$, $K_{\mathrm{I}}^{\theta} = 4.10^3$ $K_{\mathrm{P}}^{\mathrm{bR}} = 0.6$, and $K_{\mathrm{I}}^{\mathrm{bR}} = 0.2$).

$K_{\mathrm{P}}^{\theta}, K_{\mathrm{I}}^{\theta}$	$K_{\mathrm{P}}^{\mathrm{bR}}, K_{\mathrm{I}}^{\mathrm{bR}}$	$\Delta_B(10^9)$	$\sigma_B^2(10)^3$	Δ_{PSNR}	σ_{PSNR}^2
$3.10^4, 0$	$0.6, 0$	5.36	213.5	3.55	5.93
$3.10^4, 0$	$0.6, 0.2$	2.14	3.15	3.55	5.93
$3.10^4, 4.10^3$	$0.6, 0.2$	2.37	6.15	0.73	5.23

Table 5.3: System performance in terms of Δ_B, σ_B^2, Δ_{PSNR}, and σ_{PSNR}^2, when using (P_P), (P_PI), and (PI_PI) controllers, when the encoding rate is adjusted according to the buffer levels in bits while statistically multiplexing four video programs.

6.2.2 Buffer levels in seconds

In the second set of experiments, we focus on the system performance when the encoding rate is adjusted according to the buffer levels in seconds.

Parameters of the bandwidth allocation and the encoder control using P controllers and those using PI controllers are set in such a way that the corresponding roots, solutions of (5.55) using $K_{\mathrm{P}}^{\theta} = 2.10^4$, $K_{\mathrm{I}}^{\theta} = 1.10^3$, $K_{\mathrm{P}}^{\tau\mathrm{R}} = 2.5.10^4$, and $K_{\mathrm{I}}^{\tau\mathrm{R}} = 5.10^3$, are located in the unit circle, see Figure 5.20 with $\gamma = 5.10^{-5}$.

The bandwidth allocation is performed using a PI controller using $K_{\mathrm{P}}^{\theta} = 2.10^4$ and $K_{\mathrm{I}}^{\theta} = 1.10^3$ corresponding to the least average PSNR variance σ_{PSNR}^2 and discrepancy Δ_{PSNR} over the N programs. The encoder control is performed using PI controller with $K_{\mathrm{P}}^{\tau\mathrm{R}} = 2.5.10^4$ and $K_{I}^{\tau\mathrm{R}} = 5.10^3$.

The buffer levels in bits, the PSNR, the bandwidth allocation θ, the encoding rate, and the buffer levels in seconds τ of each program are represented in

- Figure 5.18, when P controllers are used in both bandwidth allocation and encoder control (P_P),
- Figure 5.19, when PI controllers are used in both bandwidth allocation and encoder control (PI_PI).

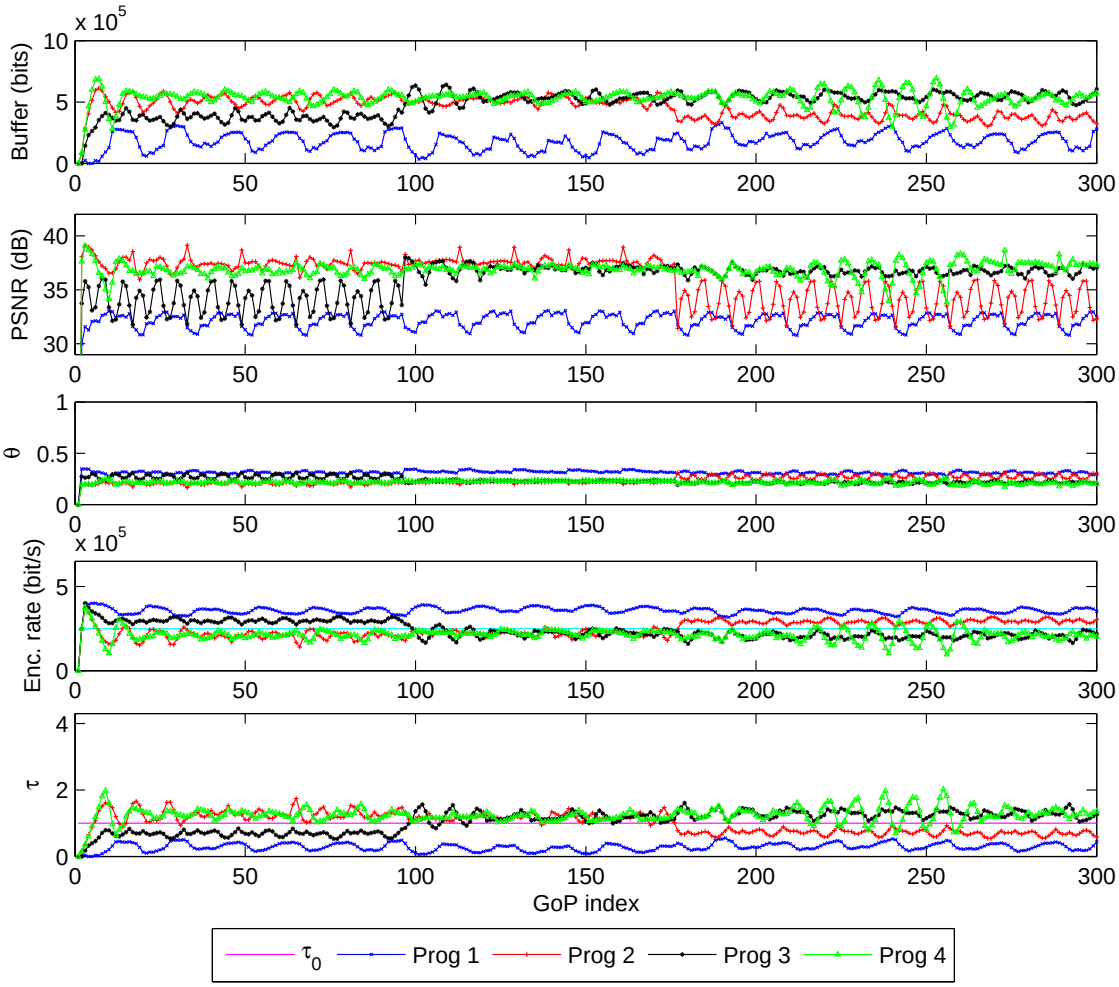


Figure 5.18: System performance using P controllers for both bandwidth allocation and encoding rate controls (P_P), when the encoding rate is adjusted according to the buffer levels in seconds while statistically multiplexing four video programs using $K_{\mathrm{P}}^{\theta} = 2.10^4$ and, $K_{\mathrm{P}}^{\tau \mathrm{R}} = 2.5.10^4$.

The PI controller allows low discrepancy between the buffer levels in seconds at equilibrium and the reference level τ_0. In addition, even in presence of scene change, the control system reaches equilibrium after short transients and satisfy the quality fairness constraint.

The system performance in terms of discrepancy of the buffer levels in seconds Δ_τ, variance of the buffer levels in seconds σ_τ^2, the PSNR discrepancy Δ_{PSNR}, and the PSNR variance σ_{PSNR}^2 are summarized in Table 5.4, when using (P_P), (P_PI), and (PI_PI) controllers. Here, as when buffer levels in bits are controlled, using the PI controller, the discrepancy of the buffer levels in seconds Δ_τ and the variance of the buffer levels in seconds σ_τ^2 decrease significantly compared to the P controller without disturbing the behavior of the PSNR control where the PSNR variance σ_{PSNR}^2 value is kept almost the same. The PI controller in the bandwidth allocation reduces the PSNR discrepancy and better satisfy the fairness constraint than when using a P controller.

Figures 5.15, 5.16, 5.18, and 5.19 show some oscillating behavior after the tran-

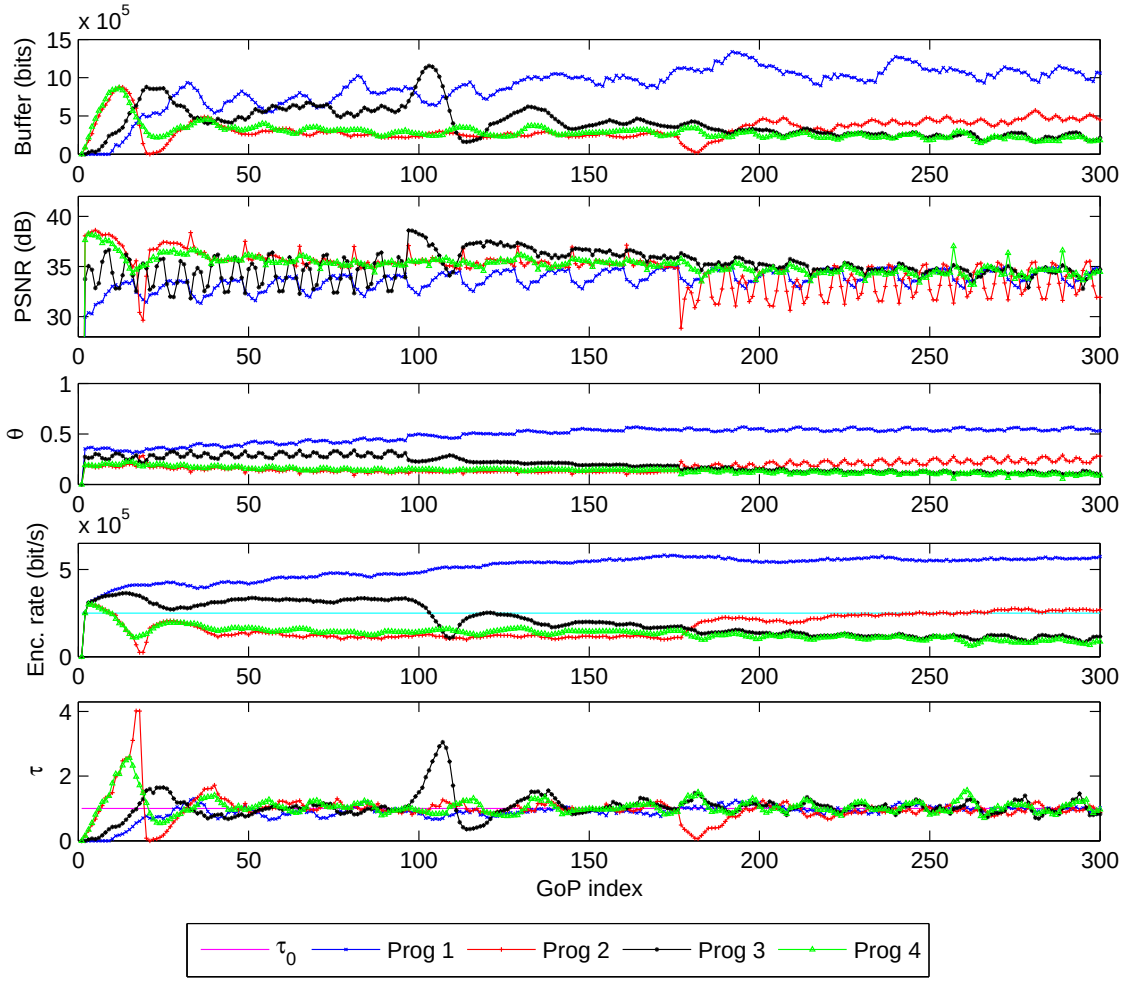


Figure 5.19: System performance using PI controllers for both bandwidth allocation and encoding rate controls (PI_PI), when the encoding rate is adjusted according to the buffer levels in seconds while statistically multiplexing four video programs using $K_{\mathrm{P}}^{\theta} = 2.10^4$, $K_{\mathrm{I}}^{\theta} = 1.10^3$, $K_{\mathrm{P}}^{\tau\mathrm{R}} = 2.5.10^4$, and $K_{\mathrm{I}}^{\tau\mathrm{R}} = 5.10^3$.

$K_{\mathrm{P}}^{\theta}, K_{\mathrm{I}}^{\theta}$	$K_{\mathrm{P}}^{\tau\mathrm{R}}, K_{\mathrm{I}}^{\tau\mathrm{R}}$	Δ_{τ}	σ_{τ}^2	Δ_{PSNR}	σ_{PSNR}^2
$2.10^4, 0$	$2.5.10^4, 0$	2	0.39	3.9	6.25
$2.10^4, 0$	$2.5.10^4, 5.10^3$	0.08	0.32	3.85	6.33
$2.10^4, 1.10^3$	$2.5.10^4, 5.10^3$	0.21	0.45	1.5	5.99

Table 5.4: System performance in terms of Δ_{τ}, σ_{τ}^2, Δ_{PSNR}, and σ_{PSNR}^2, when using (P_P), (P_PI), and (PI_PI) controllers when the encoding rate is adjusted according to the buffer levels in seconds while statistically multiplexing four video programs.

sient phase. This is due to the discrepancy between the model and the actual R-D characteristics and to the fact that the QPs provided by the optimization process have to be rounded before being used by the video encoders.

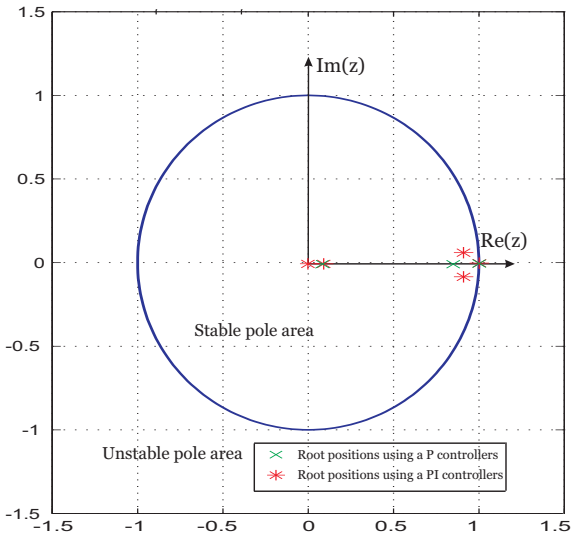


Figure 5.20: Position of the roots using (P_P) and (PI_PI) controller using video sources, when buffer levels in seconds are controlled ($K_{\mathrm{P}}^{\theta} = 2.10^4$, $K_{\mathrm{I}}^{\theta} = 1.10^3$, $K_{\mathrm{P}}^{\tau\mathrm{R}} = 2.5.10^4$, and $K_{\mathrm{I}}^{\tau\mathrm{R}} = 5.10^3$).

7 Conclusion and open issues

In this chapter, we propose a control system that allows the SM of multiple video sources in a distributed way. A MANE is in charge of buffering the encoded streams and building a centralized bandwidth allocation among programs in order to satisfy some quality fairness constraint. Buffer levels in bits or in seconds are then fed back to the rate control system of each video server to adapt the encoding rates and the video quality so that a reference buffer level is achieved. The proposed control process involves PI controllers for both bandwidth allocation and encoder control process. Experimental tests show that PI controller allows having similar video qualities and buffer levels between the multiplexed four program.

This control system can be performed on scalable video sources. The same buffer and rate control system can be used to determine the optimal encoding rate for each video server. Only the rate control algorithm has to be adapted to fit the target bit rate. This can be done by adjusting the number of scalable layers. The main difficulty comes from the fact that not all rate points may be reached in a continuous way (this would require very fine grain scalability). In a simple case, the number of scalable layers leading to a rate point closest to the requested one will then have to be applied. This is at the expense of some loss in the coding efficiency and also of larger discrepancy between the model and the actual R-D characteristics and thus, more oscillations. Increasing the number of scalable layers would lead to an additional overhead. Thus, we should focus on the trade-off between the granularity of the scalable stream (number of scalable layers) and the efficiency of the rate control algorithm. Control at the frame level should be considered, however, in this work, we have not been concerned by the communication delay problem between the MANE and the encoders. This is because we perform rate control at the GoP level with a duration of 0.5 s which is larger than typical communication delays of the order of tens milliseconds. Nevertheless, while performing rate control at the frame level, frame duration and the communication delay are of the same order of

magnitude. Thus, addressing the problem of communication delays is one of the future steps.

Part III

Scalable Video Rate Control: Stochastic Optimization

Chapter 6

Introduction to Markov Decision Process for Video Delivery

1 Introduction

This chapter presents an overview of the Markov Decision Process (MDP) framework. We introduce the several notations and algorithms required to cast cross-layer optimization problems for the delivery of multimedia contents into the MDP framework. This formalism allows deriving a long-term control policy maximizing some weighted sum of rewards.

An illustrative example of video encoding and transmission is considered throughout this chapter. The considered control problem is formulated and solved using the MDP framework. This illustrative example will be extended for video delivery control problem over wireless network considered in the following chapters. We also introduce the Reinforcement Learning (RL) technique concerned with how an agent ought to take actions in an unknown environment so as to maximize some notion of cumulative reward.

2 Markov Decision Process

2.1 Definitions

A Markov decision process (MDP) comprises an *agent* and its *environment*, interacting as in Figure 6.1. At each time step, $t = 1, 2, 3, \ldots$, the agent perceives the state of the environment, s_t, and selects an action, a_t. In response to the action, the environment changes to a new state, s_{t+1} and emits a reward, $r_{t+1} \in \mathbb{R}$. In finite MDPs the states and actions are chosen from finite sets. In this case the environment is characterized by arbitrary probabilities $P(s', s, a)$, for each possible transition from a state, s, to a next state, s', given an action, a.

More specifically, an MDP is a 4-tuple $(\mathcal{S}, \mathcal{A}, P, r)$, where $\mathcal{S}$ is the set of states of the system, $\mathcal{A}$ is the set of actions, $P\left(s_{t+1}, s_t, a_t\right)$ determines the transition probability from $s_t \in \mathcal{S}$ at time t to $s_{t+1} \in \mathcal{S}$ at time $t+1$, when the action $a_t \in \mathcal{A}$ is applied to the system. Finally $r : \mathcal{S} \times \mathcal{A} \longmapsto \mathbb{R}$ is a reward function that denotes the immediate reward for applying a certain action a_t in a certain state s_t with transition probability $P(s_{t+1}, s_t, a_t)$.

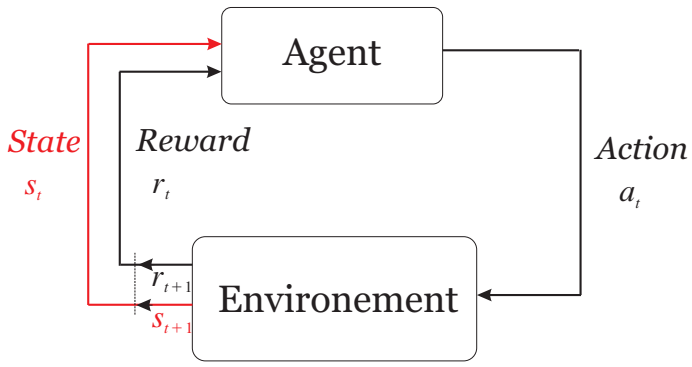


Figure 6.1: The agent-environment interaction [SB98].

To construct the transition probability matrix, a particular property of the environment and its state signals that is of particular interest is defined, called the Markov property. A state signal is said to be Markovian, or to have the Markov property is a state signal that summarizes in a compact way the past states which has lead to this situation. The environment dynamics can be defined by

$$Pr(s_{t+1} = s'|s_t, a_t, s_{t-1}, a_{t-1} \dots s_0, a_0) = Pr(s_{t+1} = s'|s_t, a_t) \tag{6.1}$$

for all s', and all possible values of the past events: $s_t, a_t, s_{t-1}, a_{t-1} \dots s_0, a_0$.

Thanks to the Markov property, the state-transition distributions and the rewards depend only on the current state and the current action, and not on the history of previous actions and states. This property allows the description of the current state containing all information relevant to choose an action in the current state.

At each time step, the agent implements a mapping from states to probabilities of selecting each possible action. This mapping is called the agent *policy*, denoted π_t, where $\pi_t(s, a)$ is the probability that $a_t = a$ if $s_t = s$. The agent goal is to maximize the total amount of reward it receives over the long run. More formally, in the simplest case, the agent should choose each action a_t so as to maximize the expected discounted reward:

$$V^{\pi}(s_t) = E_{\pi}\left[\sum_{k=0}^{\infty}(\gamma)^k r_{t+k+1}|s_t\right], \tag{6.2}$$

where the parameter $0 \leq \gamma \leq 1$ is the discount factor, which defines the relative importance of present and future rewards. The expected value is denoted by $E_{\pi}[.]$ given that the agent follows policy π. The *foresighted* policy that maximizes the above sum takes into account the impact of the current actions on the future reward. When $\gamma = 0$, only the immediate reward is maximized. The corresponding policy is called *myopic* policy, as only the immediate benefit is searched for. Immediate reward should be weighted more heavily than future rewards since, for example, typical traffic and channel dynamics only have stationary behavior over short time intervals such that the future dynamics cannot be easily predicted without error, moreover the applications lifetime is not known a priori.

The state-value function is denoted by V^{π}. An optimal policy, denoted π^*, is a policy whose values are greater than or equal to that of all other policies at all states.

The MDP framework is abstract and very flexible, allowing it to be applied to many different problems and in many different ways. In fact, the MDP framework is well adapted for the control of a system which states satisfy the Markov property. Video transmission control is one of the several examples that can be cast in the MDP framework due the Markovian behavior of the video content, the buffer state, and the transmission channel. It has been shown that video traffic can be accurately modeled as a Markov process [TC01]. Hence, MDP provides a rigorous methodology for using these existing models to improve video transmission performance, see [FvdS09a, MvdS10].

2.2 Example: Video encoding and transmission control

In this section, we cast an example of video packet encoding and transmission in the MDP framework. Our aim is to design a rate control algorithm for video encoding and transmission. We assume that the video frames are coded at the frame rate F (frames/second) and that each frame may be encoded using a quantization parameters (QP) $Q \in \{16, 23, 30, 36, 40, 42\}$. Every frame is coded as a P frame, with the exception of the first frame encoded as an I frame.

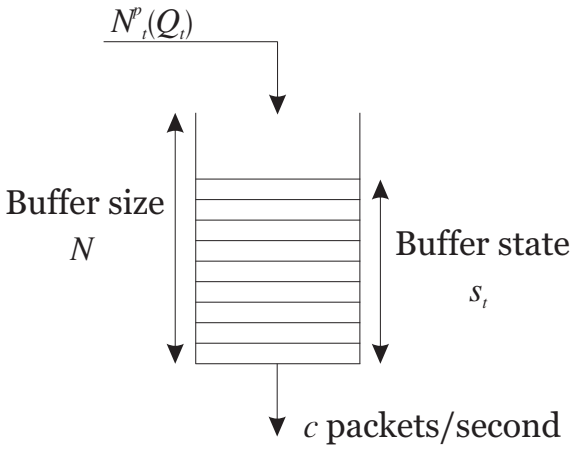


Figure 6.2: The rate control model.

Here, we assume having the relation between the average bit-rate R (in bits/s) and the quantization parameter Q, denoted as $R(Q)$ and similarly, the relation between the distortion D and the QP Q, denoted as $D(Q)$ for *Stephan.qcif* and *Silent.qcif* sequences encoded using the H.264/AVC encoder. We assume all frames are I-frames so that all frames are independent and R and D depend only on the Q of the current frame.

The encoded video frames are injected into a post-encoder buffer for potential transmission. At the same time, the packets in the buffer are transmitted at the rate c packets/second, where all packets in the buffer are considered having the same size L. The buffer occupancy at time t is measured in number of packets before injecting the control frame t into the buffer.

The state of the system consists of the buffer state s corresponding to the number of stored packet. Here, $s \in \mathcal{S} = \{1 \dots B\}$.

The action that the system can take is a corresponding to the QP to encode the next frame. Action a is defined in $a \in \mathcal{A} = \{Q_1 \dots Q_q\}$, where q is the maximum

number of QP available at the encoder.

Once states and actions are identified, one has to determine the transition probability matrix from moving from state s_t to state s_{t+1} using action a_t is

$$P\left(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t\right) = \Pr\left(\mathbf{s}_{t+1} \,|\, \mathbf{s}_t, \mathbf{a}_t\right). \tag{6.3}$$

3 Dynamic programming

Dynamic Programming (DP) refers to a collection of algorithms which can be used to compute optimal policies given a perfect model of the environment. *Bellman* (in 1957) and *Howard* (in 1960) laid the foundations for most of the early research for the DP algorithms. By exploiting the Markov property, they originated the most popular algorithms (value iteration and policy iteration) and made significant contributions to the mathematical study of MDP.

3.1 Policy iteration

We use the notation $\pi(s)$ for the action that policy π recommends at state s.

In policy iteration, a policy is first evaluated, and then improved. Policy evaluation consists of working out the value of every state s under policy π.

We refer to $V^\pi(s)$ as the value of state s and V^π as the state-value function.

By separating the first and the subsequent terms in the sum in (6.2), and using the Markov property, one gets the following Bellman equation [SB98]

$$V^\pi(s_t) = r_t(s_t, \pi(s_t)) + \gamma \sum_{s' \in \mathcal{S}} P(s', s_t, \pi(s_t)) V^\pi(s'). \tag{6.4}$$

The second term in (6.4) is the expected infinite discounted rewards that will be gained by executing that policy. For a given policy π, one obtain V^π by solving a set of linear equations. Once we know the value of each state under the current policy, we consider whether the value could be improved using

$$\pi^*(s_t) = \arg\max_{a \in \mathcal{A}} \left\{ r_t(s_t, a) + \gamma \sum_{s' \in \mathcal{S}} P(s', s_t, a) V^\pi(s') \right\}, \tag{6.5}$$

by changing the considered action.

If it can, we change the policy to take the new action whenever it is in that situation. When no improvements are possible, then the policy is optimal. The policy iteration algorithm is detailed in Algorithm 1.

3.2 Value iteration

One drawback to policy iteration is that each of its iteration involves policy evaluation, which may itself be a long iterative computation. If policy evaluation is done iteratively, then convergence to the optimal value function may take a long time. In fact, the policy-evaluation step of policy iteration can be truncated in several ways without loosing the convergence of policy iteration. This algorithm is called *value iteration*.

Algorithm 1 Policy iteration algorithm

Input: an MDP tuple $(\mathcal{S}, \mathcal{A}, P, r)$
Input: an accuracy threshold θ
Output: $\pi(s)$
Initialize π_0 randomly
$k \leftarrow 0$
  Repeat
  Initialize V_0^{π} randomly
  $i \leftarrow 0$
    Repeat
      For all states $s \in \mathcal{S}$ **do**
      $V_{i+1}^{\pi_k}(s) \leftarrow \sum_{s' \in \mathcal{S}} P(s', s, \pi_k(s)) \left[r_t(s, \pi_k(s)) + \gamma V_i^{\pi_k}(s) \right]$
      End for
    $i \leftarrow i + 1$
    Until $\left|\left| V_i^{\pi_k} - V_{i-1}^{\pi_k} \right|\right| \leq \theta$
    For all states $s \in \mathcal{S}$ **do**
    $\pi^*(s)_{k+1} \leftarrow \arg\max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} P(s', s, a) \left[r_t(s, a) + \gamma V_i^{\pi_k}(s') \right]$
    End for
  $k \leftarrow k + 1$
  Until $\pi^*(s)_{k+1} = \pi^*(s)_k$

The Q-function $Q(s, a)$, denotes the expected value of performing action a at state s. Initially, Q can be set to zero, the set of action-value function $Q : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is updated simultaneously according to the formula

$$Q(s_t, a_t) = r(s_t, a_t) + \gamma \sum_{s_{t+1} \in \mathcal{S}} P(s_{t+1}, s_t, a_t) \max_{a' \in \mathcal{A}} Q(s_{t+1}, a'). \tag{6.6}$$

Two loops on all states and actions are performed in order to collect all action-value functions using reward values generated at each state-action pair. Then, the optimal value function V^* is obtained from

$$V^*(s_{t+1}) = \max_{\mathbf{a} \in \mathcal{A}} (Q(s_{t+1}, a)). \tag{6.7}$$

The value iteration algorithm is detailed in Algorithm 2.

3.3 Example: Solution

Reward: The delay experienced by the packets of frame t is defined as the difference between the time at which the data is injected into the post-encoder buffer and the time at which the data is transmitted. We define the reward associated with each frame as

$$r_t(s_t, a_t) = \lambda \underbrace{\left(\frac{1)}{F} - \frac{s_t + N_t^p(Q_t)}{c} \right)^+}_{\text{Delay reward}} + \mu \underbrace{(B - s_t - N_t^p(Q_t))^+}_{\text{Overflow reward}} - \underbrace{D_t(Q_t)}_{\text{Distortion reward}}, \tag{6.8}$$

where λ and μ are positive parameters to trade-off the incurred delay, distortion and penalize buffer overflow, $N_t^p(Q_t)$ is the number of packets in frame t, $(x)^+ =$

Algorithm 2 Value iteration algorithm

Input: an MDP tuple $(\mathcal{S}, \mathcal{A}, P, r)$
Input: an accuracy threshold θ
Output: $\pi(s)$
Initialize V_0 randomly
$i \leftarrow 0$
  Repeat
    For all states $s \in \mathcal{S}$ **do**
    $V_{i+1}(s) \leftarrow \arg\max_{a\in\mathcal{A}} \sum_{s'\in\mathcal{S}} P(s', s, a') \left[r_t(s, a) + \gamma V_i(s')\right]$
    End for
  $i \leftarrow i + 1$
  Until $\left|\left|V_i^{\pi_k} - V_{i-1}^{\pi_k}\right|\right| \leq \theta$
  For all states $s \in \mathcal{S}$ **do**
  $\pi(s) \leftarrow \arg\max_{a\in\mathcal{A}} \sum_{s'\in\mathcal{S}} P(s', s, a) \left[r_t(s, a) + \gamma V_i(s')\right]$
  End for

$\max(0, x)$, and B is the maximum number of packets allowed in the buffer. We define $N_t^p(Q_t)$ as follows:

$$N_t^p(Q_t) = \left\lceil \frac{R(Q_t)}{L} \right\rceil, \tag{6.9}$$

where $\lceil . \rceil$ denotes rounding toward $+\infty$ and L is the packet length ($L = 20$ kbits for Silent sequence and $L = 45$ kbits for Stephan sequence).

Buffer state evolution: The buffer state transition from frame t to frame $t + 1$ is given by

$$s_{t+1} = \min\left\{ \left(s_t + N_t^p(Q_t) - \frac{c}{F} \right)^+, B \right\}, \tag{6.10}$$

where $\frac{c}{F}$ is the number of departing packets, and B is the buffer size. The time step refers to the fixed frame duration $\frac{1}{F}$.

Objective: The objective in this problem is to maximize the discounted reward with the discount factor γ, $(0 \leq \gamma \leq 1)$, *i.e.* $\sum_{k=0}^{\infty} (\gamma)^t r_t$, by determining the optimal QP to be used for each frame Q_t.

Problem resolution: To solve the optimization problem described in the video encoding and transmission control example in Section 2.2, we use value iteration algorithm to find the optimal value function $V^* : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ which satisfies (6.7).

Experimental tests are conducted using Stephan and Silent sequences in CIF format. The impact of the myopic and foresighted policies for their actions on achieved cumulative expected rewards are quantified.

We compute the optimal myopic policy $\gamma = 0$, and the foresighted policy $\gamma = 0.9$ using the $R(Q)$ and $D(Q)$ measures obtained using different quantization parameters $Q \in \{16, 23, 30, 36, 40, 42\}$ with a frame rate $F = 15$ frames/second for 100 frames of the Silent and Stephan sequences. In these experiment tests, we assume that for a particular QP the frame will always have the average bit rate and average distortion associated with that QP in both Stephan and Silent sequences.

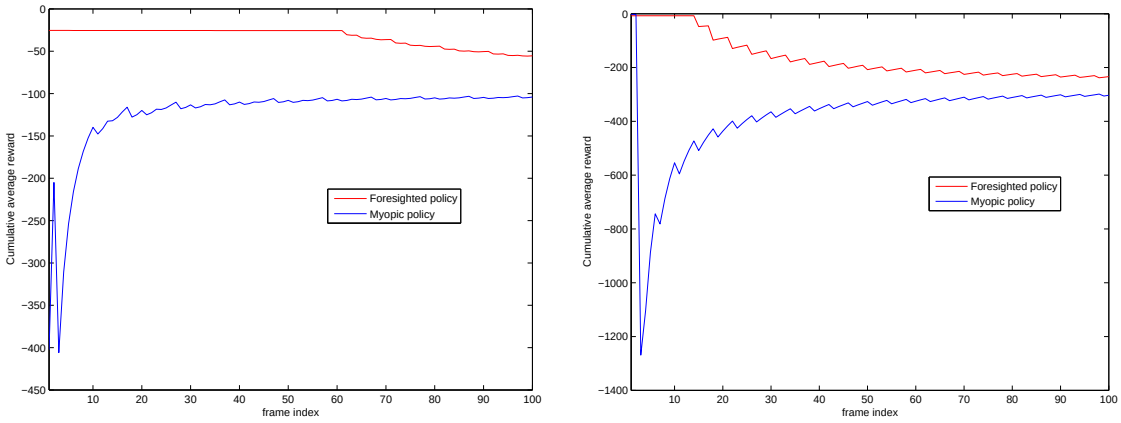


Figure 6.3: Cumulative average reward achieved using both myopic and foresighted policies for Stephan (a) and Silent(b) sequences.

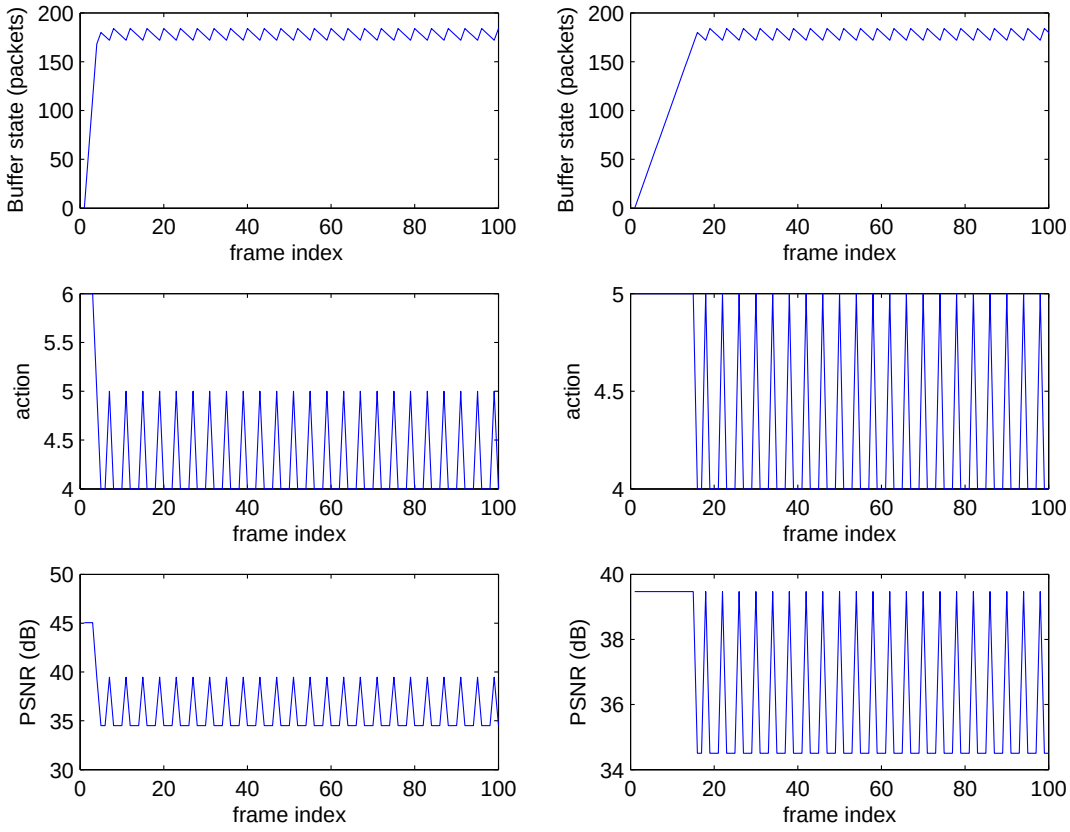


Figure 6.4: States, actions and PSNR evolution, using myopic and foresighted control for the Silent sequence.

The buffer is considered initially empty $s_0 = 0$. The draining rate from the buffer, considered constant at each time slot is $c = 200$ packets/second, the buffer size is $B = 200$ packets, and the trade-off parameters are set to $\lambda = 0.5$ and $\mu = 100$. The considered values of λ and μ allows the incurred delay, the distortion and the buffer overflow rewards, considered in the general reward function in (6.8), having the same order of magnitude at equilibrium.

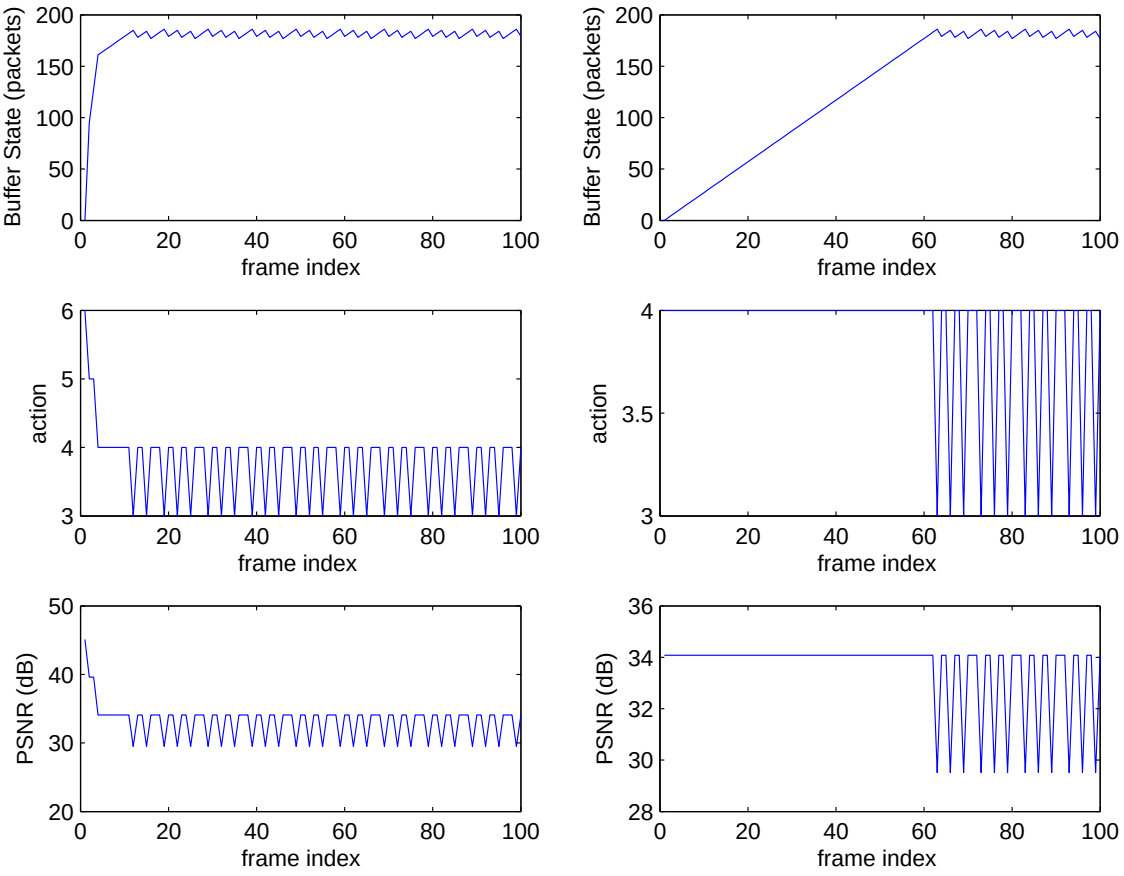


Figure 6.5: States, actions and PSNR evolution, using myopic and foresighted control for Stephan sequence.

In Figure 6.3, the cumulative average reward $c(r)$ achieved using both myopic and foresighted policies for Stephan (a) and Silent(b) sequences are illustrated with

$$c(r_t) = \frac{\sum_{k=0}^{t} r_k}{k}. \tag{6.11}$$

From Figure 6.3, one can see that the foresighted policy leads to the best performance in terms of the cumulative rewards. This means that the considered optimization problem using foresighted policy allows a higher reward in average than when using myopic policy.

In Figures 6.4, and 6.5, we plot the myopic and the foresighted policies (*i.e.*, the QP that should be selected in each buffer state), the buffer state evolution and the generated PSNR for the Silent and Stephan sequences. Figure 6.3 illustrates the cumulative average reward versus time, where the reward is defined as in (6.8) using both myopic and foresighted policies for Stephan and Silent sequences.

We can see that the obtained PSNRs using the foresighted policy are better in average than those obtained using the myopic policy, since the actions determined by the myopic policy does not consider the expected rewards. Nevertheless, since the reward function is expressed as a function of the distortion but also as a function of the buffer evolution reward maximizing the reward function does not mean that this will results in a higher received PSNR at all times. Thus, it is better to express the

reward function in (6.8) as a function of the PSNR only to provide better received PSNR when using the foresighted policy.

In this example, we studied the optimization problem cast into the MDP framework under the assumption that the dynamics of the environment is completely known. The issue in this case is just the relative efficiency of various ways of computing optimal policies. Nevertheless, in various cases, the environment dynamics are partially or completely unknown. Reinforcement learning (RL) allows solving the control problem without assumptions on the dynamics of the environment. A wide variety of intermediate cases have also been studied in RL and optimal control.

4 Reinforcement Learning

Reinforcement learning (RL) is the study of how artificial systems can learn to optimize their behavior in an unknown environment using rewards and punishments. In this section, we introduce and describe the different elements required to efficiently use RL techniques. We also provide the key elements of the mathematical structure while using RL method.

4.1 Definitions

In RL, we study the problem faced by an agent that must learn behavior through trial-and-error interactions with a dynamic environment. The learner and decision-maker is called the agent. The environment includes any element outside of the agent with which it interacts. In RL, agents interact with their environment and use their experience to choose or avoid certain actions based on their consequences. Actions that led to high rewards in a certain situation tend to be repeated whenever the same situation recurs, whereas choices that led to comparatively lower rewards tend to be avoided.

Initially, the agent does not know what effect its actions will have on the next state it will visit, nor what immediate reward this actions will produce. It particularly does not know what action is best to do. Thus, he tries out various actions at various states, and gradually learns which one is best at each state so as to maximize its immediate reward or a discounted sum of rewards.

4.2 Learning the optimal policy

In practice, the reward and transition probability functions may be (partially) unknown a priori. The problem at hand is estimating the value functions in this situation. The reason we want to estimate these value functions is so that they can be used to accurately choose an action that will provide the best possible total reward, after being in that given state. RL approach can be used to learn the optimal action-value function Q^* and the optimal policy π^* on-line, without first estimating the unknown reward and transition probability functions.

4.2.1 Q-learning

Q-learning [SB98] is an RL technique that works by learning an action-value function that gives the expected reward of taking a given action in a given state and following a fixed policy thereafter. One of the strengths of Q-learning is that it is able to compare the expected reward of the available actions without requiring a model of the environment. Q-learning algorithms work by estimating the values $Q(s,a)$ of state-action pairs (s,a). Once these values have been learned, the optimal action from any state is that one with the highest Q-value. After being initialized, Q-values are estimated on the basis of experience as follows:

- At the current state s_t, select an action a_t. This leads to an immediate reward r_t, and arrival at a next state s_{t+1}.
- Update $Q(s,a)$ based on the Q-learning rule as follows

$$Q_t(s_t,a_t) \leftarrow Q_t(s_t,a_t) + \alpha_t \left(r_t + \gamma \max_{a' \in \mathcal{A}} Q_t(s_{t+1},a'_t) - Q_t(s_t,a_t) \right), \quad (6.12)$$

where s_{t+1} is the resulting state in $[t+1, t+2)$, which is distributed according to $p(s_{t+1}|s_t,a_t)$; a'_t is the greedy action in state s_{t+1}, which maximizes the current estimate of the action-value function; $\alpha \in [0,1]$ is a time-varying learning rate parameter. The ϵ-greedy selection method corresponds to using most of the time the best available action, but in ϵ % of the time an action is randomly selected from the set of possible actions. $Q_0(s,a)$ can be initialized as an identically zero initial Q-function for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, see Algorithm 3

Algorithm 3 Q-learning algorithm

Initialize $Q(s,a)$
  For each time step **do**:
   choose a from s using policy derived from Q (ϵ-greedy)
   Take action a, observe r', s'
   $Q(s,a) \leftarrow Q(s,a) + \alpha\left[r' + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$
  End for
$s \leftarrow s'$;

This algorithm converges to the correct Q-values with the probability one if the environment is stationary and depends on the current state and the action taken in it.

4.2.2 Sarsa

The Sarsa algorithm [SB98] is a learning algorithm. The major difference between Sarsa and Q-Learning, is that the maximum reward for the next state is not necessarily used for updating the Q-values. Instead, a new action, and therefore reward, is selected using the same policy that determined the original action. The name Sarsa actually comes from the fact that the updates are done using the quintuple (s,a,r,s',a'). Where: s, a are the original state and action, r is the reward observed

Algorithm 4 Sarsa algorithm

Initialize $Q(s, a)$ arbitrary
  For each time step **do**:
    choose a from s using policy derived from Q (ϵ greedy)
  Take action a, observe r', s'
    choose a' from s' using policy derived from Q (ϵ greedy)
    $Q(s, a) \leftarrow Q(s, a) + \alpha\left[r' + \gamma Q(s', a') - Q(s, a)\right]$]
  End for
$s \leftarrow s'$ $a \leftarrow a'$;

in the following state and s', a' are the new state-action pair. The Sarsa algorithm is detailed in Algorithm 4

Sarsa allows learning the value of the policy that is used to make decisions. The value functions are updated using results from executing actions determined by some policy. Contrarily to the Sarsa algorithm which updates the estimated value functions using hypothetical actions, those which have not actually been tried, the Q-learning methods updates the value function based strictly on experience.

4.2.3 Exploration and Exploitation

A trade-off between exploitation and exploration must be made. If a learner exploits too much, it will be blind to possibly better solutions which could result in higher rewards, while a learner which keeps on exploring the environment in a fully random way will end up accumulating very little reward.

An action selection method in which all information is exploited without ever trying new paths is called a greedy action selection strategy. The drawback of such a method is clearly that it gets easy stuck in sub-optimal solutions. A simple but effective variant of greedy action selection methods is the ϵ-greedy selection method.

5 Conclusion

In this chapter, we introduced several notations and tools required to cast control of multimedia distribution chain into the MDP framework. A video encoding and transmission optimization example is also considered in this chapter. The obtained performance in terms of expected reward using the considered example, highlights the advantages using foresighted policy compared to myopic one. Nevertheless, the system performance highly depends on the definition of the reward function. In fact, the reward function includes the several constraints to be satisfied, however, maximizing it does not systematically imply that all constraints will be satisfied. Thus, one must define the reward function so that its maximization allows the optimization objectives to be satisfied. In addition, in the considered illustrative example, several assumptions concerning the frame dependency, the RD models and the channel rate are considered. These assumptions will be taken into account in the next chapter, where we consider an end-to-end scalable video transmission optimization problem that addresses the problem of video on demand delivery over a time-varying wireless channel cast into the MDP framework.

Chapter 7

Cross-layer Optimization of Scalable Video Delivery over Wireless Channel

1 Introduction

In this chapter, we addresses the problem of video-on-demand delivery over a time-varying wireless channel. A filtering algorithm among scalable layers is considered in charge of filtering the scalable layers according to the channel and system conditions to maximize the average received video quality. The filtering algorithm is based on Quality of Experience (QoE) criteria, namely maximum PSNR and minimum playback margin at the receiver buffer to prevent freezing of the decoded video in bad channel conditions. The Markovian behavior of the video content, of the buffer state, and of the transmission channel justifies a solution in the framework of Markov Decision Processes (MDP). This framework allows deriving a foresighted control policy maximizing some long-term discounted sum of rewards. Experimental results illustrate the benefits of this approach compared to a short-term (myopic) policy. The proposed layer filtering process has led to the submission of a patent application [CSK10c] and then to a publication in the proceeding of ACM Multimedia 2010 [CvdSsk10].

1.1 Challenges

Efficient video streaming is considered as one of the most important and challenging application for next generation wireless networks [AFK+11]. Current infrastructures are not prepared to deal with the increasing amount of video traffic. The current Internet, and in particular the mobile Internet, was not designed with video requirements in mind and, as a consequence, its architecture is very inefficient for handling video traffic. Enhancements are needed to cater for improved QoE and improved reliability in a mobile network which provide dynamically varying resources with only limited support for the Quality of Service (QoS) required by the delay-sensitive, bandwidth-intense and loss-tolerant multimedia applications. One of the key challenges associated with the reliable and efficient multimedia transmission over wireless networks is the dynamic characteristics of both the wireless networks

and multimedia sources.

As presented in [ZAPS+07], the video distortion in a wireless system depends on the distortion resulting from the available bit-rate to encode the video, denoted by D_{enc}, and on the quality of the wireless channel (information loss due to transmission errors), denoted by D_{loss} so that the total distortion of the decoded video is

$$D_{\text{dec}} = D_{\text{enc}} + D_{\text{loss}}. \tag{7.1}$$

Due to the specific set of delay constraints of a streaming video traffic and in order to reduce D_{dec}, the transmitter should determine for each video packet the transmission/encoding rate and its scheduling time. At the receiver, the incoming video packets are received and stored in the decoder buffer. The decoder reads video packets from this buffer and displays the video sequence in real-time. Real-time display means that once the receiver begins displaying the received video, the display process continues uninterrupted, without stalling. If video data does not arrive on time to be displayed, then this data is considered lost. To ensure this continuous playout, a playback margin is considered where frames are stored in the display buffer before playback starts reducing the impact of delay jitter, losses and variations of the network characteristics on video quality.

The objective is to maintain the playback buffer level around a certain reference level. This can be achieved by adjusting the source and channel rates so that the probability of playback-buffer starvation is very small. Generalizing this target to the issue of multiuser streaming system (multi-severs/multi-users) is an even more challenging task.

Typically, in case of wireless shared channels, the multiuser dimension is captured by the scheduling policies used at the MAC layer. The users are served according to some criteria related to the channel quality, the fullness of the queue, and the flow priority. The MAC scheduler element is responsible for deciding which users in the cell are going to transmit or receive data and for allocating the most adequate set of resources to be used. In LTE systems, for example, all traffic is packet switched and all channel resources are shared among all users in the cell. As a consequence the scheduling algorithm is critical to guarantee performance levels and QoS requirements.

The objective of this chapter is to propose a scheduling scheme based on a joint source control and radio link buffer management for N-unicast video streaming over wireless shared channels. All users having a video traffic in a cell are optimized jointly.

In a wireless network, see Figure 7.1 video delivery control can be done at different places (at the encoder output, in the core network, at the scheduler, at the receiver side, etc.).

Controlling the video stream in the network so that the target objectives are satisfied is a challenging task mainly due to the fact that video encoder parameters are difficult to adjust and due to the time-varying channel conditions which come with delay to the control unit.

These challenges require video optimization solutions that can enable the service provider to determine network conditions, lay down policy rules with respect to content access, tune video quality, and enhance the QoE at the receiver end.

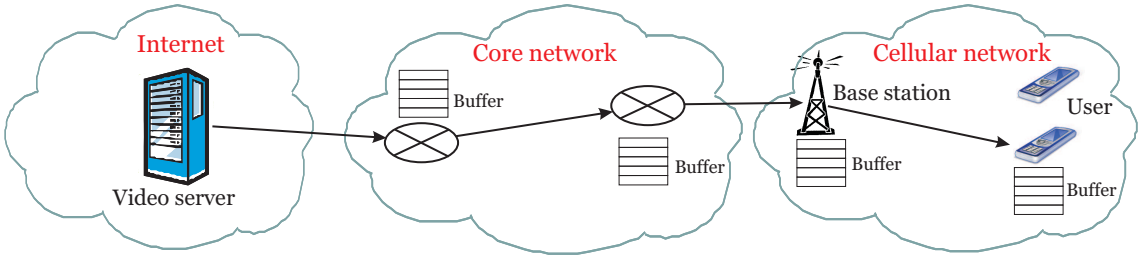


Figure 7.1: Video streaming architecture.

1.2 Related works

Existing wireless networks provide time-varying resources with only limited support for the QoS required by delay-sensitive, bandwidth-intense and loss-tolerant multimedia applications. To overcome these challenges, packet scheduling optimization has been extensively investigated in recent years in order to maximize the quality of the decoded multimedia stream given the underlying resource constraints. A brief summary of the previous results on video delivery is provided in Table 7.1

In [CYOK99], the problem of delay-sensitive rate control for transmission of real-time video over burst-error wireless channels is considered using Lagrangian optimization of the encoding rate. The quantization parameters of the source coder are optimized dynamically to maximize the quality while satisfying expected rate constraints using accurate model of the rate-distortion (R-D) characteristics of the source. Buffers at encoder and receiver side allow mitigating some mismatch due to feedback delays.

A QoE based optimization framework for multi-user wireless video delivery is proposed in [Ta10]. Transcoding and packet dropping are used in the rate adaptation scheme by investigating their impact on user perceived video quality in presence of constrained transmission resources. The algorithm allows a network operator handling multiple video streams with various contents and selecting dynamically an appropriate rate adaptation scheme to be applied to each video stream.

Since the capacity of a wireless channel varies randomly with time, controlling the video transmission in wireless networks using deterministic optimization tools may result in discrepancies between the actual system states and the modeled one. One can resort to optimization techniques that account for the uncertainty of the considered environment. For example, stochastic optimization, where probability distributions characterize uncertain problem parameters, is an approach that has been applied widely. Within the stochastic optimization, the problem of video scheduling can be performed in a foresighted (long term) fashion, which not only consider the immediate impact of encoding or scheduling decisions, but also their long-term impact.

In [COR02], a Markov channel model, and a model of the source are considered to optimize the quantization parameters of the non-scalable video coder. The problem is formulated into the MDP framework and an optimal control policy is evaluated off-line for each possible state of the channel and of the source, which allows a low-complexity control law.

An opportunity for easily adapting the source rate is provided by scalable video coding techniques [SMW07]. Such coders generate compressed streams organized

Method	Perf. Metric	Codec	Channel model	Control policy	Traffic characteristics	Feedback delay	Solution
[CYOK99]	Dist.	H.261	N-SMC	QP	Delay constraint	Delayed	ALO
[COR02]	Dist.	H.263	2-SMC	QP	Delay constraint	Delayed	MDP
[LEP+03]	Energy	MPEG-4	N-SMC	QP and Trans. rate	Delay and quality constraint	Immediate	ALO
[dCR03]	Dist.	MPEG-4 FGS	N-SMC	Scheduling and error concealment	Playback delay and smooth quality	Immediate	MDP
[KHZ+03]	Dist.	MPEG-4 FGS	N-SMC	Scheduling	Delay and packet loss constraint	Immediate	NLO
[HAK04]	Output bit-rate	Scalable MPEG-4	Gilbert-Elliot	Encoding rate	Playback delay	Immediate	NLO
[NKWR07]	Dist.	H.264 AVC	Predicted link error model	Scheduling: Number of RLC PDUs	Delay deadline	Delayed	State machine
[Ta10]	MOS	H.264 AVC	WINNER model [ESBa05]	QP and packet drop	No	Immediate	NLO
[MK10]	Dist.	H.264 SVC	2-SMC	Rate allocation and layer scheduling	Frame dependency and drift effect	Immediate	NLO
[FVdS10]	Dist.	H.264 SVC	N-SMC	Scheduling	Delay constraint and packet dependency	Immediate	MDP
[CHJBdV11]	Dist.	H.264 SVC	N-SMC	Scheduling	Playback delay	Immediate	MDP
Chapter 7	Dist.	H.264 SVC	2-SMC	Layer filtering	Playback delay and application buffers control	Immediate	MDP
Chapter 8	Dist.	H.264 SVC	N-SMC	Layer filtering	Frame dependency and joint MAC and application buffers control	Delayed	MDP

Table 7.1: Summary of of various existing research on video delivery over wireless channels: MOS (Mean Opinion Score), NLO (Lagrange optimization with numerical solution), ALO (Lagrange optimization with analytic solution), N-SMC (N-State Markov Chain), Dist. (distortion).

in spatial, temporal, and quality scalability *layers* from which several substreams may be extracted to get reconstructed videos at various resolution, frame rates, or achieving various R-D compromises. In the context of streaming of scalable audio contents, [MO00] proposed a distortion optimized strategy for transmission over a lossy network. Outdated packets and packets containing low-priority (enhancement) layers may be dropped to leave more space to packets with high-priority (base) layers.

Buffering at the client is the principal means via which we can allow a minimum playback margin and attempt to maintain efficient subjective video quality. Thus, the receiver buffer management strategy must be set in order to be adapted to the changing channel conditions and satisfy a minimum playback margin constraint to keep smooth video quality. The scheme proposed in [HAK04] controls the probability of starvation of the playback buffer while smoothing out the rate variations at the output of the source coder. The system parameters are computed to maximize the source bit rate subject to a starvation probability constraint. The difficulty with scalable video coders is the drift propagation due to the loss of some high level layers at the receiver side. An estimation algorithm is proposed in [MK10] to determine the impact of enhancement layer truncation, drift/error propagation, and error concealment. The available transmission rate, the possibly time-varying channel conditions, the hierarchical prediction structure of the SVC, and the possibility of random packet losses are taken into account to design a scheduling technique for scalable video packets based on distortion estimation technique.

In [FVdS10], the problem of scheduling delay-sensitive media data for transmission over time-varying wireless channels is formulated into the MDP framework. The heterogeneous multimedia data characteristics and time-varying channel conditions are explicitly considered.

1.3 Main contributions

In order to address the several challenges discussed before, we start in this chapter by considering a 1-unicast video transmission scheme over a wireless channel.

Our concern is to improve the quality of the received video based on the end user experience by minimizing the video distortion and providing a continuous playout at the client. We propose a control system, located in a proxy in the core network, that performs layer filtering of a scalable encoded stream stored in some post-encoder buffers. This system controls both the layer filtering taking into account the priority order between the layers as well as various buffer levels. Time-varying channel conditions are considered. The layer filtering process is cast in the context of MDP, where, at each time, the process should take an action that allows maximizing the received quality and satisfying the system constraints using myopic or foresighted policies. Controlling the receiver buffers by introducing constraints on their levels allows providing some minimum playback margin.

2 Scalable layer filtering process for video delivery over a wireless channel

This section focuses on video quality optimization in the context of video transmission applications to end user while considering quality, channel and buffering constraints.

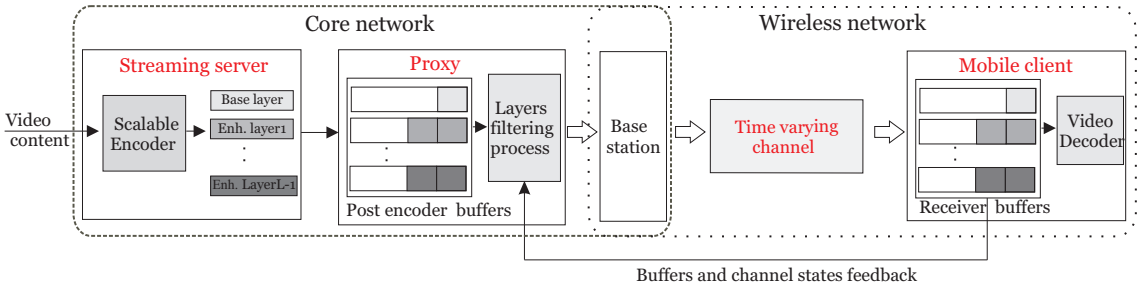


Figure 7.2: End-to-end video transmission scheme.

A video transmission system to mobile user consists of several components, see Figure 7.2. One may identify the video source, the source coder, the wired part of the network, buffers spread at several places within the network, a wireless front-end, the wireless channel, and the receiver. The receiver is typically is mobile client. Nevertheless, in this work, the handover issue is not addressed. This means that we assume that the streaming server transmit video to a mobile receiver considered linked to the same base station during the streaming session. Video encoded using scalable coder are considered. Layer filtering and buffer management are jointly considered to adapt to the changing channel conditions. The layer filtering is performed in a proxy in the core network but it can also be performed in the base station. If some parts in the transmission scheme are uncontrolled (the source, the channel), most other parts may be tuned to satisfy the system requirements. The time-varying characteristics of the source and of the channel make the end-to-end optimization of such transmission chain quite difficult.

The filtering process is in charge of choosing at each time the number of scalable layer to transmit per frame so that the received quality is maximized and a minimum playback margin is satisfied. In this problem, each SVC layer in the encoded stream corresponds to an SNR level leading to an encoded stream with different quality levels. Each SNR layer of the encoded stream is stored in its corresponding buffer in the proxy and then in its corresponding receiver buffer. Satisfying the target constraints requires an efficient control of the level of the buffers of each SNR layer in both the proxy and the receiver.

2.1 Video transmission system

2.1.1 Streaming server

In the streaming server, the video sequence is segmented into frames and encoded into L layers: a base layer and a set of $L-1$ enhancement layers. Frames are generated with a constant period of time T and is identified by its temporal index t.

Among the different type of scalability, we consider here the SNR scalability. The encoding parameters (quantization step, frame rate, etc.) are controlled by the streaming server, independently of the remainder of the chain. The proxy is assumed not to control these parameters.

2.1.2 Proxy

The L SNR layers are packetized by the streaming server and fed, via an idealized wired network (lossless), to the L post-encoder buffers in the proxy. Having one buffer per layer allows identifying each stored packet corresponding to each SNR layer. This way of storing packets allows an easy differentiation of the actions applied to each layer. A layer filtering process is considered in the proxy in charge of choosing the number of scalable layer to transit. The control is performed at each time step t corresponding to the frame duration.

Unlike most of the existing scheduling systems where packets are transmitted or dropped, here, the layer filtering process may decide to send or to drop packets, but it may also choose neither of these two actions an choose a wait action. The wait action allows not dropping the packet immediately when the channel is in bad condition and storing it longer in the post-encoder buffer if there is enough space. In this chapter, a packet contains a whole SNR layer of a frame and so its size is not fixed.

In addition, constraints on the available bandwidth have to be satisfied. For that purpose, the layer filtering process in the proxy exploits some feedback from the receiver to estimate the channel conditions. In order to limit the delay introduced by the transmission of feedback information, the proxy is placed close to the base station.

2.1.3 Base station and channel

The base station in cellular network contains usually additional buffers for each user connected to it. It is responsible to perform the rate and bandwidth allocation (MAC scheduling, see [FL02, AH05]) among users. In this work, the delay induced by these buffers at MAC layer is neglected. Moreover, both base station and transmission channel are modeled by a two-state Markov model.

2.1.4 Receiver

The receiver hosts the video decoder and one buffer per scalability layer. The level of the receiver buffers and the state of the channel are fed back to the proxy with a period T, and are assumed to arrive without delay nor error. Outdated packets are dropped, without being decoded. Some packet-loss concealment may be put at work at receiver side.

2.2 Markov decision process formulation

In this section, the problem of designing an optimal layer filtering policy of L SNR scalable layers over a wireless channel is translated in the framework of discrete-time MDP.

Thus, one has to identify for the considered system all components of the tuple $(\mathcal{S}, \mathcal{A}, P, r)$, see Section 2 in Chapter 6, where $\mathcal{S}$ is the set of states of the system, $\mathcal{A}$ is the set of actions, $P\left(s_{t+1}, s_t, a_t\right)$ determines the transition probability from $s_t \in \mathcal{S}$ at time t to $s_{t+1} \in \mathcal{S}$ at time $t+1$, when the action $a_t \in \mathcal{A}$ is applied. Finally r is a reward function that denotes the immediate reward for applying a certain action a_t in a certain state s_t with transition probability $P(s_{t+1}, s_t, a_t)$.

2.2.1 States

The set of states of the system consists of the state h_t of the base station and the channel, the states s^e_ℓ, $\ell = 1 \ldots L$, of the post-encoding buffers hosted by the proxy, and the states s^r_ℓ, $\ell = 1 \ldots L$, of the receiver buffers. The vector gathering all state at time t is $s_t = (h_t, s^e_t, s^r_t) \in \mathcal{S}$.

2.2.1.a Channel model

A large literature dealing with the representation and analysis of burst error channels using simple Markov models is available. The classical 2-state Gilbert-Elliott model [Gil60, Ell63] for burst noise channels has been widely used and analyzed. In [WM95], a finite-state Markov channel (F-SMC) was built by partitioning the received instantaneous signal-to-noise ratio (SNR) into intervals. The relation between a physical channel and its finite-state Markov model for a packet transmission system is established in [ZK99].

We assume a N_h-state Markov model to describe the behavior of the MAC buffer and the channel.

At time t, the state h_t, assumed constant within the interval defined by two consecutive time indexes $t-1$ and t, represents a rate within the set $\mathcal{R}^c = \{R^c_0, \ldots, R^{c_{N_h-1}}\}$ expressed in bits/s.

State transition probability $p_{x,k} = p(s^h_t = x \,|s^h_{t-1} = k)$ from moving from state $x \in \{1, \ldots, n\}$ to state $k \in \{1, \ldots, n\}$ are considered known *a priori* but can be estimated on-line.

2.2.1.b Buffers

The states of the ℓ-th post-encoder and receiver buffer, with $\ell \in \{1, \ldots, L\}$, are denoted by $s^e_\ell \in \mathcal{S}^e_\ell$ and $s^r_\ell \in \mathcal{S}^r_\ell$. They represent the level in number of packets of the corresponding buffer. The vectors of states of all post-encoder and receiver buffers are respectively denoted by $\mathbf{s}^e_t = (s^e_{1,t}, \ldots, s^e_{L,t}) \in \mathcal{S}^e$ and $\mathbf{s}^r_t = (s^r_{1,t}, \ldots, s^r_{L,t}) \in \mathcal{S}^r$ with $\mathcal{S}^e = \mathcal{S}^e_1 \times, \ldots, \times \mathcal{S}^e_L$ and $\mathcal{S}^r = \mathcal{S}^r_1 \times, \ldots, \times \mathcal{S}^r_L$.

Various granularity levels may be considered to represent the content of a buffer. One may consider the number of bits it contains, independently of the number of packets. One may alternatively consider that all packets have the same size and track the number of such packets stored in the buffer as in the example considered in Section 2.2 in Chapter 6, see also [MvdS09] and [FvdS09a], this approach is considered for MAC packets which have usually a constant size. This approach does not allow an easy tracking of the number of frames in a buffer, since each layer in a frame may be encoded in packets with variable size. Thus, one may finally consider the number of packets corresponding to a layer of a frame, as proposed in [MvdS10].

This state representation facilitates the control of the transmission delay. The main drawback of this approach is that it does not allow keeping a precise track of the content in bits of each buffer, which has thus to be assumed large enough.

The latter representation in the context of filtering of scalable layers would lead to a very large number of values taken by the state. Value iteration algorithms would not converge in a reasonable amount of time. Thus, a coarser representation of the level of the state has to be considered. One has essentially to control the level of the buffers. For example, at receiver side, buffer underflow, especially of the high-priority layer, may lead to *frozen* display, due to a lack of packets.

Thus, buffer level are quantized to get three state values, namely, underflow, satisfying occupancy, overflow. Receiver buffers are considered large enough so that overflow state is not considered.

Thus, the values taken by the states are further quantized to get $\mathcal{S}_\ell^e = \{\mathrm{U}, \mathrm{G}, \mathrm{O}\}$ and $\mathcal{S}_\ell^r = \{1, 2\}$ with $\ell = 1 \dots L$, with

- State value U corresponds to a buffer underflow, *i.e.,* when the actual level is below S_{min}^x, with $x \in \{e, r\}$.
- State value G indicates a good occupancy, with an actual number of packets between S_{min}^e and S_{max}^e for the post-encoder buffer and above S_{min}^r for the receiver buffer.
- State value O corresponds to a buffer overflow, only considered for the post-encoder buffer, when its actual level is larger than S_{max}^e.

Thus, the state space of the post-encoder buffers $\mathcal{S}^e = \{0, 1, \dots, B^e\}$ is quantized into N_p intervals to model each post-encoder buffer state and the state space of the receiver buffers $\mathcal{S}^r = \{0, 1, \dots, B^r\}$ into N_r intervals to model the receiver buffer state. B^e and B^r are respectively the maximum number of packet that can be stored in each of the post-encoder buffers and in each of the receiver buffers. The quantized state transition probability is expressed as follows

$$p(s_{t+1} \in [i_1, i_2] | s_t \in [j_1, j_2], \mathbf{a}_t) = \frac{\sum_i \sum_j p(s_{t+1} = i | s_t = j, \mathbf{a}_t) p(s_t = j)}{\sum_j p(s_t = j)},$$

where $[i_1, i_2]$ and $[j_1, j_2]$ are two of the several intervals and i and j are elements in these intervals. In case of state quantization, we assume states to be uniformly distributed within the different intervals leading to

$$p(s_{t+1} = j \in [j_1, j_2]) = \frac{1}{j_2 - j_1}. \tag{7.2}$$

2.2.2 Actions

The layer filtering process has to determine the number of packets from each layer to send to the channel. The action $a_{\ell,t}$ taken for the ℓ-th layer at time t represents the number of transmitted packets from the post-encoder buffer, when its value is positive, or the number of dropped packets when it is negative. If $a_{\ell,t} = 0$, packets are neither transmitted nor dropped.

Since a layer may be decoded only if the corresponding higher-importance layer has already been decoded, when some layer is dropped from the post-encoder buffer, all refinement layers of the dropped layer belonging to the same frame should be dropped too.

The vector gathering all actions at time t is denoted by $a_t{=}(a_{1t}, \ldots, a_{Lt}) \in \mathcal{A}$, where $\mathcal{A}$ is the set of actions.

2.2.3 Transition matrix

Once all states and actions have been identified, one has to determine the transition probability matrix $\Pr(\mathbf{s}_{t+1} \,|\, \mathbf{s}_t, \mathbf{a}_t)$ denoted by $P(\mathbf{s}_t, \mathbf{s}_{t+1}, \mathbf{a}_t)$, where $\mathbf{s}_t = (\mathbf{s}_t^e, \mathbf{s}_t^r, h_t)$, assuming that the transitions from $\mathbf{s}_t$ to $\mathbf{s}_{t+1}$ only depends on the value of $\mathbf{s}_t$ and of $\mathbf{a}_t$.

The transition of the post-encoder states only depends of the actions. The transition of the receiver state only depends on the action and of the channel state. Finally, the channel state transition depends on the previous value of the state. Thus, $P(\mathbf{s}_t, \mathbf{s}_{t+1}, \mathbf{a}_t)$ may be factorized as follows

$$P(\mathbf{s}_t, \mathbf{s}_{t+1}, \mathbf{a}_t) = \underbrace{p\left(\mathbf{s}_{t+1}^e | \mathbf{s}_t^e, \mathbf{a}_t\right)}_{\text{Encoding state transition}} \underbrace{p\left(\mathbf{s}_{t+1}^r | \mathbf{s}_t^r, \mathbf{a}_t, h_t\right)}_{\text{Receiver state transition}} \underbrace{p\left(h_{t+1} | h_t\right)}_{\text{channel state transition}} . \quad (7.3)$$

The encoding state transition and the receiver state transition are calculated using first the non-quantized state transition that depends on the previous state and the chosen action, and then the derived quantized state probability in (7.2). The channel state transition are considered known, these probabilities can be learned and updated on-line.

2.2.4 Reward function

One has to define the reward function that the optimization process should maximize in order to maximize the PSNR of the decoded frame. Ideally, one should estimate and maximize the PSNR of the decoded frame. However, it is difficult to efficiently estimating it at the controller due to the delay between the playback instant and the decision instant. Thus, we introduce a reward function at the control unit so that requirements are satisfied.

1. The quality of the decoded video should be maximized.

2. The available rate should be efficiently used.

3. The level of the post-encoder buffers should be adjusted so that under and overflow situation are avoided.

4. The receiver buffers should be managed in order to provide some playback margin against the unavailability of the channel.

To take all these requirements into account, at time t, the following reward function is considered

$$r_t(\mathbf{s}_t, \mathbf{a}_t) = \underbrace{\sum_{\ell=1}^{L} \gamma_l a_{\ell,t}}_{\text{transmission reward}} + \underbrace{\beta\nu \left(R_t^T(\mathbf{a}_t, h_t) - R_t^c\right)}_{\text{bandwidth constraint reward}} + E\left[\underbrace{\sum_{l=1}^{L} \lambda_l \rho(s^e_{\ell,t}, a_{\ell,t})}_{\text{enc. buffer reward}} + \underbrace{\sum_{\ell=1}^{L} \mu_\ell \rho(s^r_{\ell,t}, a_{\ell,t})}_{\text{rec. buffer reward}}\right]. \tag{7.4}$$

The positive parameters γ_ℓ, λ_ℓ, μ_ℓ, with $\ell = 1 \ldots L$, and β help to trade-off the importance of the various constraints imposed on the system. The reward function (7.4) involves several parts corresponding to the several system requirements. These parts are now briefly detailed.

2.2.4.a Buffer rewards

Here, the post-encoder and the receiver buffers constraints are addressed. A satisfying buffer level should correspond to a positive reward, whereas overflow or underflow should result in a negative reward. Thus, a reward function $\rho(.)$ defined as follows is introduced.

$$\begin{cases} \rho(s_t, a_t) = 1 & \text{if } s_{t+1}{=}2, \\ \rho(s_t, a_t) = -1 & \text{if } s_{t+1} \neq 2, \end{cases} \tag{7.5}$$

where s_{t+1} is the state obtained at $t+1$ after applying action a_t when the state of the buffer at time t is s_t. $\rho(.)$ is then used to calculate the expectations in (7.4) for the buffer state reward evaluated over all possible states of the buffers at time $t+1$.

Taking values for the parameters λ_ℓ and μ_ℓ that depend on the layer l allows giving more importance to the satisfaction of buffering constraints for the base layer compared to the refinement layers.

2.2.4.b Transmission reward

We assume that the only information kept about a packet is the SNR layer to which it belongs, this may, *e.g.*, be obtained from the RTP header [SCFJ03]. Accounting for more information, *e.g.*, an indication that the packet belongs to an Intra or an Inter-encoded frame would probably improve the efficiency of the system, but this would be at the price of a significant increase of complexity. In addition, the impact of packet loss concealment at receiver side is not taken into account.

Based on the assumption that increasing the amount of transmitted packets increases the received quality, the transmission reward should help in maximizing the amount of transmitted packets. The parameters γ_l allow giving a higher priority to packets belonging to the base layer compared to those of the enhancement layers. The relative importance of each layer has to be adjusted with respect to the increase in quality it provides.

When a packet is dropped, a negative reward equal to the opposite of the reward obtained by transmitting the packet is considered.

2.2.4.c Bandwidth constraint reward

The bandwidth constraint reward is defined as

$$\nu\left(R_t^T(\mathbf{a}_t, h_t) - R_t^{\mathrm{c}}\right), \tag{7.6}$$

where $\nu(\cdot)$ is a function defined as follows

$$\nu(x) = -\mathrm{abs}(x) - \nu_0 \mathrm{sgn}(x), \tag{7.7}$$

with $\mathrm{abs}(x)$ is the absolute value of x, $\mathrm{sgn}(x) = 1$ if $x > 0$, $\mathrm{sgn}(x) = -1$ if $x \leq 0$, and ν_0 is some positive constant.

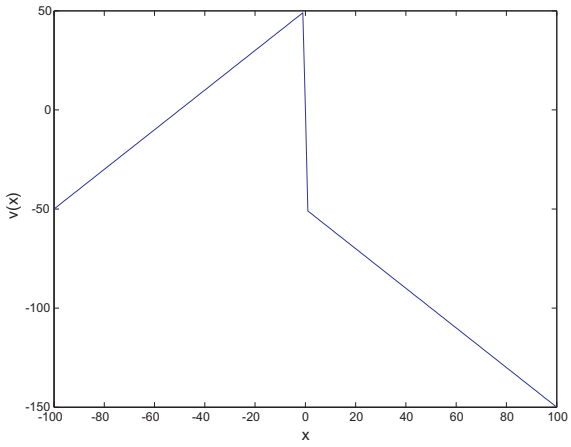


Figure 7.3: Representation of $\nu(x)$.

The function $\nu(\cdot)$ introduces a strong penalty when $R_t^T(\mathbf{a}_t, h_t) - R_t^{\mathrm{c}}$ is larger than zero, corresponding to a required rate larger than the instantaneous channel transmission rate. When it is small, a positive reward is provided, which decreases when $R_t^T(\mathbf{a}_t, h_t)$ is too far from R_t^{c}, see Figure 7.3 for $x = [-100 \ldots 100]$ bits/s and $\nu_0 = 50$ bits/s.

Note that this bandwidth constraint reward does not eliminates the possibility of temporarily transmitting at a rate higher than the channel transmission rate. This effect is mitigated by buffers at MAC layer present in the base station. This situation should however be exceptional. This is why it is strongly penalized.

Assuming that the channel state h_t is known, the transmission reward as well as the channel constraint reward are deterministic.

2.3 Experimental results

2.3.1 Simulation conditions

The performance of the proposed layer filtering process is evaluated using *Foreman.qcif* at a frame rate $F = 30$ fps. Experiments are performed using the H.264/SVC encoder (JSVM 9.11) [VWS07]. The period at which the control system is operating is $T = 1/F$. The video sequences is encoded and divided into GoPs of $N_G = 16$ frames.

2.3.1.a Video encoder

The Foreman sequence is encoded using three MGS scalability layers per frame ($L = 3$) corresponding to different video qualities. In the proposed filtering process, dropping some of the enhancement layers may introduce some drift. In order to minimize the drift, all frames are encoded as key pictures [SMW07]. With this configuration, motion compensation is only performed using the base layer of the previous frame as reference and thus any loss of a packet containing an enhancement layer has no impact on the motion compensation loop. This reduces the coding efficiency, but results in a coding scheme which is much more robust to losses.

Table 7.2 reports the characteristics of the three layers in terms of cumulated bit rate (expressed in kbit/s) and associated Peak Signal-to-Noise Ratio (PSNR) of the luminance component (Y PSNR) (expressed in dB).

	bit rate (kbit/s)	Y PSNR (dB)
Quality Id 0	34.67	28.67
Quality Id 1	107.01	31.51
Quality Id 2	326.99	35.82

Table 7.2: *Foreman.qcif* encoded with three MGS layers, cumulated rate and PSNR of the luminance (Y PSNR) when the decoding is performed up to the considered layer.

The evaluation of the transmission rate is performed by taking average values of the rates in each layer.

2.3.1.b Channel

A time-varying channel is considered in the simulations. It consists of a two-state Markov model ($N_h = 2$), which state switched with a period T. The instantaneous channel rate are $R_0^c = 0$ bit/s, and $R_1^c = 300$ kbit/s. The channel state transition are governed by the following transition probabilities $P(1|1) = 0.9$ and $P(0|0) = 0.7$ considered known.

2.3.1.c Buffers

The size of the post-encoder and receiver buffers and state bounds considered in our experimental test are reported in Table 8.3.

B^e	B^e_{max}	S^e_{min}	S^r	S^r_{min}
20	19	6	30	13

Table 7.3: Level at which the post-encoder and receiver buffers are considered in underflow or overflow state.

State transition probabilities for the buffer are estimated off-line from the average rate characteristics of each layer of the considered video sequences. A more accurate state transition probability can be considered using probability distribution of the encoding rate instead of the average rate.

2.3.1.d Rewards

The parameters in (7.4), given in Table 7.4, have been adjusted manually in order to trade-off the importance of the various constraints.

γ_1	300	λ_1	200	μ_1	300
γ_2	150	λ_2	100	μ_2	150
γ_3	60	λ_3	40	μ_3	60

Table 7.4: Values chosen for the parameters in the reward function.

Parameters β and ν_0 of the bandwidth constraint reward are respectively set to 0.1 (dimensionless) and 5 kbits/s to give more weight to the bandwidth constraint compared to other constraints. Values of the parameters are set in a way that the different components in (7.4) are balanced at equilibrium.

We consider four possible actions per layer: $\mathcal{A} = \{-1, 0, 1, 2\}$. The action $a_{l,t} = \{1, 2\} \in \mathcal{A}$ represents the number of transmitted packets from the post-encoder buffer, or the number of dropped packets when $a_{\ell,t} = -1$. If $a_{\ell,t} = 0$, packets are neither transmitted nor dropped.

2.3.2 Results

2.3.2.a Myopic policy versus foresighted policy

The performance obtained with a myopic policy (discount factor $\gamma = 0$) is compared to that obtained with a foresighted policy (discount factor $\gamma = 0.9$).

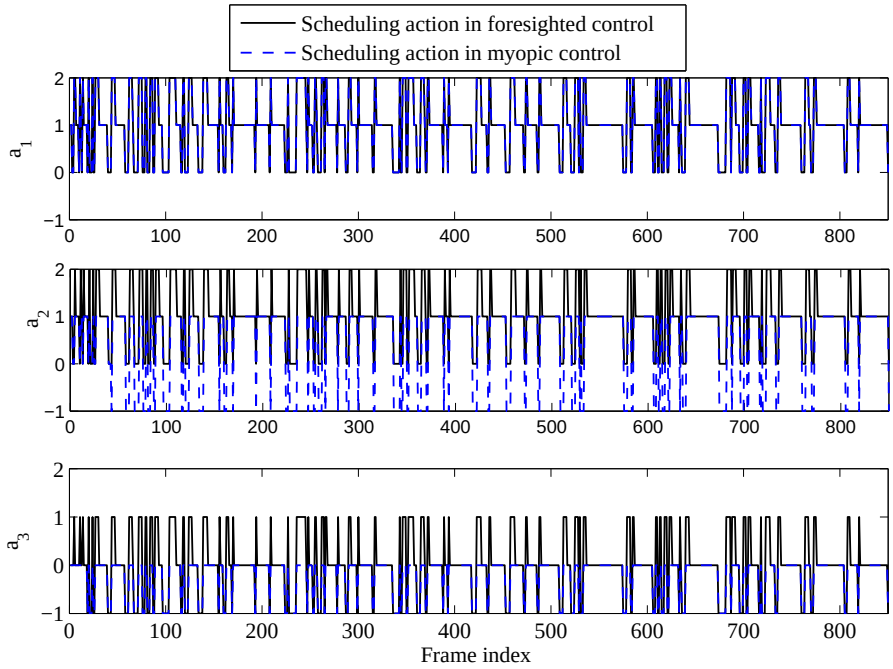


Figure 7.4: Actions for the three layers obtained with the myopic and the foresighted policies.

Figure 7.4 represents the actions for each frame at each SNR layer. These actions are generated by the proposed layer filtering process in such a way that they

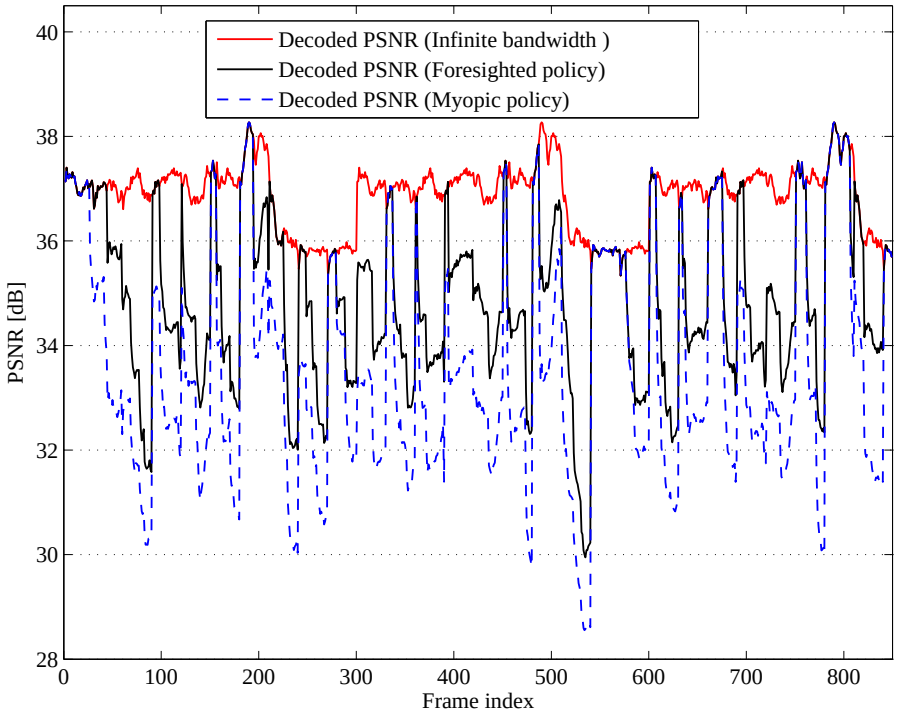


Figure 7.5: PSNR of the decoded sequences with the myopic and the foresighted policies.

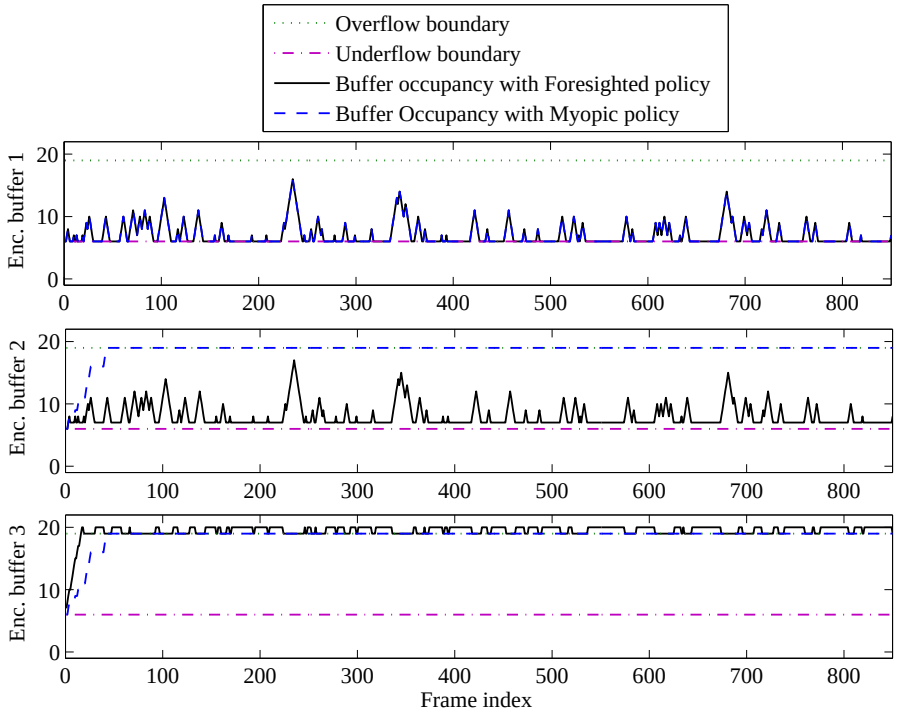


Figure 7.6: Post-encoder buffer level for the myopic and the foresighted policies.

maximize the immediate reward (myopic policy) or an infinite sum of discounted rewards (foresighted policy). For the base layer, the actions obtained do not depend on the adopted policy. For the enhancement layers, the foresighted policy manages

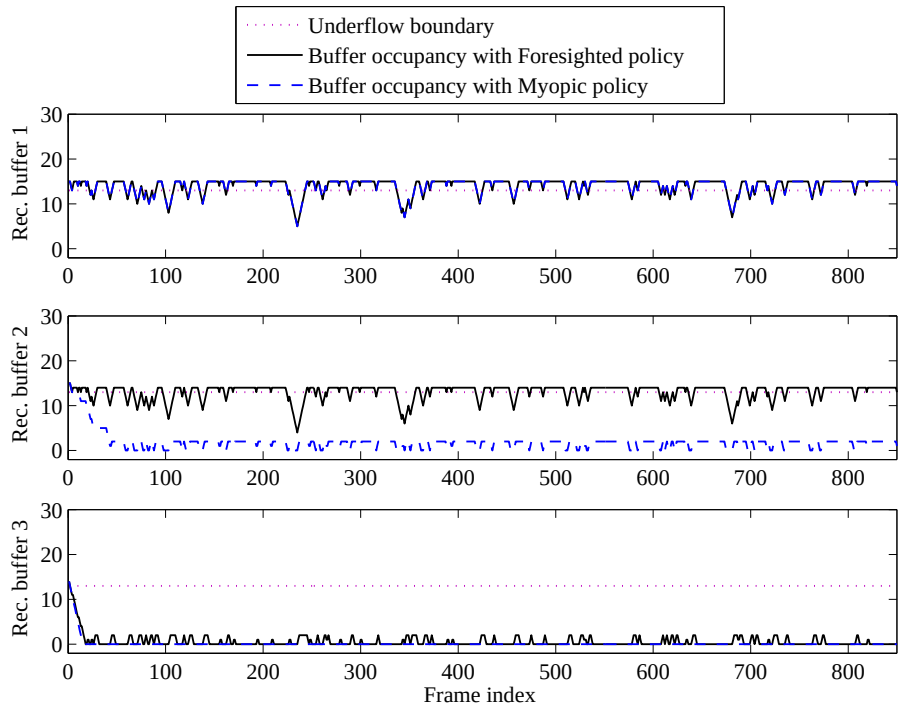


Figure 7.7: Receiver buffer level for the myopic and the foresighted policies.

better the packets transmission than the myopic policy by avoiding, when possible, drop actions at the first and at the second enhancement layers.

With the foresighted policy, 20.35% of the actions for the second enhancement layer are drop actions (-1) compared to 20.95% in the case of a myopic policy. For what concerns the first enhancement layer, no drop actions are obtained by using the foresighted policy compared to 20.35% for the myopic one.

The evolution of the PSNR for the luminance of the decoded video streams for both strategies are represented in Figure 7.5. In average (using Foreman sequence), a gain of about 0.9 dB is obtained with the foresighted policy compared to the myopic one. This gain is mainly due to more packets of the first enhancement layer reaching the receiver.

The state of the post-encoder and receiver buffers are depicted in Figure 7.6 and Figure 7.7, respectively. The post-encoder buffers are in average more often in a satisfying level with the foresighted approach than with the myopic one. The level of the post-encoder buffer for the first enhancement SNR layer is stable and the under/overflow states are avoided.

The reward function is set in a way that packets are dropped from the post-encoder buffer when the latter is in overflow state and the channel is in a bad state. Then, by keeping the post-encoder buffer in the good state, packets are more likely to be sent which results in a better reconstructed video quality. In Figure 7.7, the receiver buffers are more often at a satisfying level with the foresighted policy than with the myopic policy.

Figure 7.8 shows three versions of Frame 476: decoded using all layers, decoded with the packets provided thanks to the foresighted policy, and decoded with the packets provided by the myopic policy. A 2.25 dB improvement in PSNR is achieved using the foresighted policy compared to the myopic policy. This gap is also visible

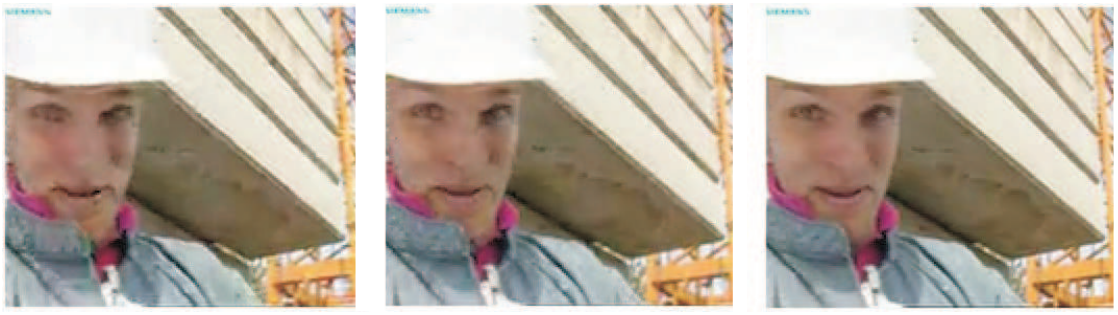


Figure 7.8: Frame 476 of foreman decoded using all layers (right, PSNR=37.06 dB), decoded with the packets provided thanks to the foresighted policy (middle, PSNR=32.6 dB), and decoded with the packets provided by the myopic policy (left, PSNR=30.35 dB).

in Figure 7.8. With both myopic and foresighted policies, the quality is not at its maximum level due to channel bandwidth constraints.

2.3.2.b Considering various channel conditions

In order to assess the robustness of the control low of the considered layer filtering process for various instantaneous rate of the channel R_1^c, additional simulations have been performed with different values of R_1^c using the same reward trade-off parameters.

In order to assess the dependency of the proposed policies to the instantaneous rate of the channel R_1^c, additional simulations have been performed with different values of R_1^c using the same reward trade-off parameters. Table 7.5 compares the myopic and foresighted policies for different values of the channel rate.

	Average transmission rate (kbits/s)		Average PSNR (dB)	
R_1^c (kbits/s)	Myopic	Foresight.	Myopic	Foresight.
200	99.84	132.42	33.77	34.63
250	115.38	156.09	34.30	35.24
300	185.46	225.93	35.49	36.42
350	211.33	225.93	35.95	36.49
400	226.10	226.10	36.49	36.49

Table 7.5: Channel use and PSNR comparison between myopic and foresighted strategies for different channels, characterized by their instantaneous transmission rate in good state.

The foresighted policy improves the average PSNR at low values of the instantaneous channel rates. The available resources are thus better exploited with the foresighted approach. At high channel rates, both policies provide similar results.

2.3.2.c Receiver buffer constraint

This section provides simulations comparing two scenario, with and without taking the receiver buffers into account. In fact, controlling the state of the receiver buffers

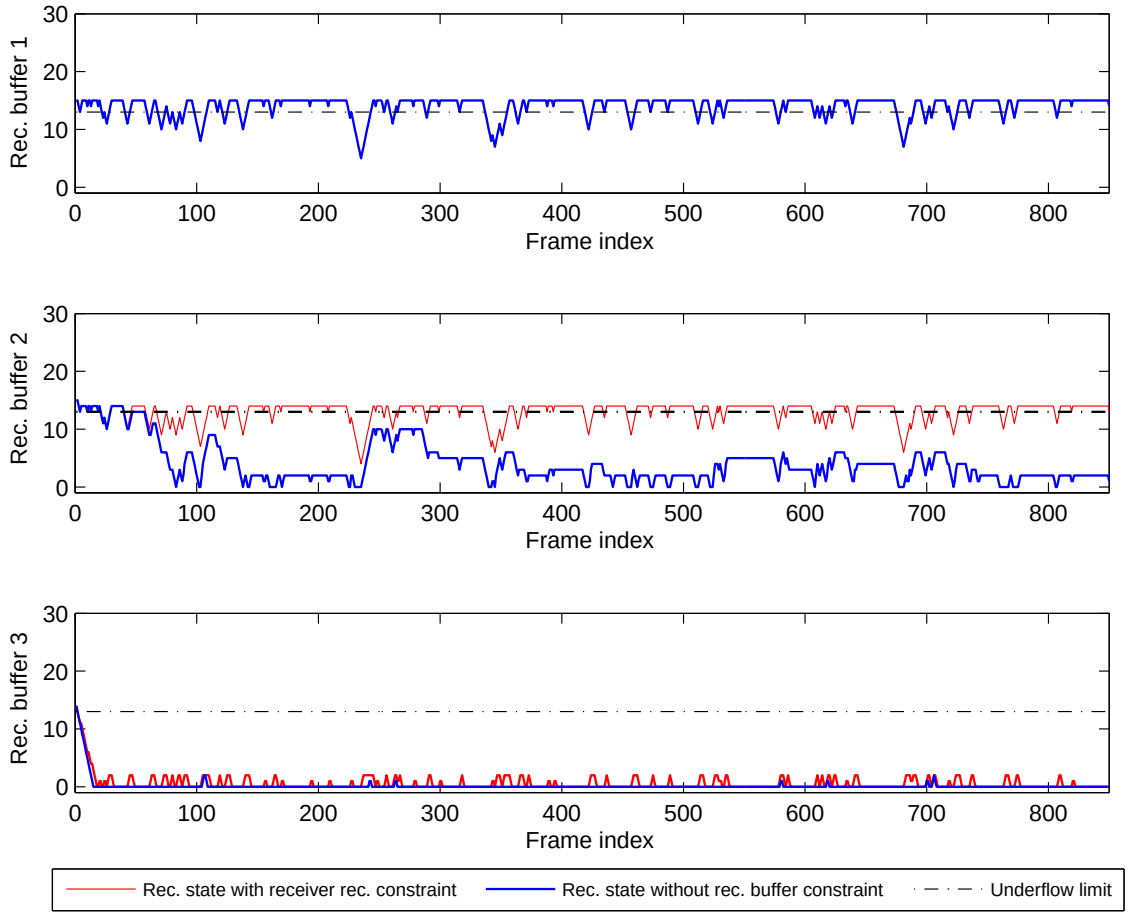


Figure 7.9: Receiver buffer level with and without receiver buffer taken into account to derive the foresighted policy.

may avoids undesirable situation, mainly receiver buffer underflow, which leads to video freezes and significant degradation of the decoded video quality. The same encoding scheme described above is used. The control process without receiver buffer is considered by setting $\mu_\ell = 0$ for $\ell = 1 \ldots 3$. Figure 7.9 shows the receiver state for each decoded frame using the foresighted approach. Accounting for the presence of the receiver buffer, allows frames to be decoded with more SNR layers in average than without taking it into account. In average, an improvement of 0.62 dB in PSNR is obtained using foresighted policy when the receiver buffer state is considered compared to without receiver buffer state case.

3 Conclusion

In this chapter, we addressed the problem of scalable encoded video transmission over a time-varying wireless channel. This problem is cast in the framework of Markov Decision Processes. Experimental results using Foreman sequence show that the foresighted approach provides an improvement of 0.9 dB in average PSNR compared to the myopic approach. In addition the buffer constraints are also better satisfied with the foresighted policy than with the myopic one. Simulation results with and without playback margin constraint at the receiver buffer show an increase

of the video quality when the playback margin is imposed. Robustnesses of the control low of the layer filtering process with respect to the channel rate and to different video sequence has still be addressed.

The proposed filtering process is performed using an off-line evaluation of the control low. However, the characteristics of the encoded videos as well as the channel state are time-varying. These information should be learned on-line to tackle the dynamic of the system variation. Optimal policies may also have to be reevaluated from time-to-time depending on the change of the complexity of the encoded video and the channel conditions. Reinforcement learning techniques can be used to directly update the state-value function and policy after encoding each frame. This technique is considered in the next chapter in order to better adapt the proposed filtering process to a more realistic environment characterized by a time-varying conditions.

In the proposed system, the channel and the receiver buffer state are used at the layer filtering process where they are assumed to be available without delay. However, in a real network these information may reach the controller with a certain delay. Video transmission optimization in delayed environment is a challenging topic. Such delay may cause instability issues. To overcome this problem, we propose in the next chapter a layer filtering control system while considering various levels of knowledge of the state of the channel namely, known state, known state with delay, and unknown state in order to achieve a more feasible solution. Moreover, the delay induced by buffers at MAC layer is not considered in the transmission system considered in this chapter. This delay will be tackled in the next chapter by introducing the MAC buffer state in the system states.

Chapter 8

On-line Learning for Scalable Video Filtering over Wireless Networks with Delayed Feedback

In Chapter 7, we proposed an optimization process for cross-layer optimization of the scalable videos delivery over wireless channels. The control process involves the channel state and the receiver buffer state information. This information is assumed immediately available at the controller, however, one challenging task when designing such control schemes is that channel state feedback comes with delay.

In order to tackle the problem of delayed feedback of the channel and of the receiver buffer state used in the previous chapter, we propose cross-layer control mechanisms to control the level of queues at the MAC layer and at the Application layer of the transmitter side *only* able to efficiently estimate the PSNR of the decoded frames. Feedback at the MAC layer is implicitly used, but no feedback at the Application layer, which may include information about channel and receiver buffer states, is considered, avoiding delayed measurements. Various levels of knowledge of the state of the channel are considered namely, known state, known state with delay, and unknown state. Without channel state observation, the control has to rely on the observation of the level of the MAC buffer only. We formulate the wireless video transmission problem as a Markov Decision Process (MDP) that explicitly considers the cooperation at the application layer and the medium access control layer, the heterogeneity of the video data, and the varying network conditions.

The proposed adaptive scalable layer filtering process for video transmission over wireless networks based on MAC buffer management has led to the submission of a patent application [CSK10d] and ten published in the proceeding of IEEE ICASSP 2011 [CMvdS+11].

1 Introduction

In this chapter, we address the problem of video on demand delivery over a time-varying wireless channel. A filtering algorithm among scalable layers is considered in charge of filtering the scalable layers, according to the channel and system conditions to maximize received video quality.

The video sequence is often encoded and encapsulated into multiple Video Units

(VU), which can be video frames, SNR layers, packets, etc. Different VUs often have different distortion impacts, delay deadlines and dependencies. A scheduling method for scalable video streams that takes into consideration the heterogeneity of the video source (frame type, SNR layer priority, delay deadline) as well as the time-varying behavior of the wireless channel is proposed in this chapter. The problem is cast in the MDP framework where a layer filtering process, considered at the transmitter side, is in charge of choosing actions and collecting rewards, related to the PSNR of decoded frames, associated to these actions. We consider essentially the problem of delay concerning the availability of the channel state at the transmitter/controller side.

1.1 Related works

Using information feedback to the controller, [KHZ+03] describes a scheduling schemes for time-varying channel conditions. A cross-layer scheduling architecture for video transmission is presented; the time-varying characteristics of the wireless channel are modeled by a discrete-time Markov model, which state is fed back to the controller. Scalable video coders have been considered in [dCR03], focusing on the application layer where an MDP framework is considered to design a long-term control policy accounting for the level of the playback buffer. In [BHKH04, LJSB04, ON07] and [YAB09], focus is more on the MAC layer, since these papers address buffer management problems within the Radio Access Network (RAN). Application packets may be dropped depending on their priority and on the level of the buffers at the MAC layer.

The main difficulty when designing such control schemes is that feedback comes with delay. In addition, frame level scheduling at the transmitter may introduce some discrepancies between the estimated PSNR reward and the actual decoded PSNR. In this chapter, we focus on both feedback delay and reward estimation delay.

Concerning the feedback delay, when considering unicast applications, various types of feedback from the receiver may be obtained, for example at the application layer via RTCP feedback [SC03] to get information about the level of buffers at application layer or at MAC layer via HARQ ACK/NACK [STB09] about the channel conditions. This delay may be of the order of tens to hundreds of milliseconds for HARQ ACK/NACK messages to one or several seconds for RTCP packets.

Concerning the reward estimation delay, at each regulation time, a reward value should be calculated for each transmission action. This reward should indicate to the control system how good or how bad the chosen action is. Nevertheless, in video transmission systems, the best reward should be the decoded PSNR at the receiver corresponding to an action that was already chosen in the past due to the presence of buffers in the communication chain. In addition, the time difference between the transmission instant and the display instant can be constant or variable, depending essentially on the considered scheduling policy. In fact, if the scheduling is performed packet by packet as in [FVdS09b] or GoP by GoP as in [FVdS10, MK09], the delay is constant and the reward estimation can be performed using some of the existing methods, dealing with constant delay MDP. For example, some prior works have explored the area of delayed environments in a number of control

problems [AN92, BW99]. Several important theoretical results have been developed for MDP with partially observable states [Son78]. The main result thus far is that an MDP with delays may be reformulated as an MDP without delays. Relative to the original MDP with delay, the MDP without delays has an augmented state space, the same state-transition structure, and a more complex reward structure involving a conditional expectation. In [WNLL09] the problems of learning and planning in Markovian environments with constant observation and reward delay is considered. A Model Based Simulation algorithm is proposed for planning in such environments and a model-based reinforcement learning is used to extend this approach to the learning setting in both finite and continuous environments. Nevertheless, if the scheduler can transmit more than one VU (frames or GoP) at each time slot, the delay between transmission and display time is variable and the estimation of the decoded PSNR becomes much more complicated. The problem of reward function with random delay is addressed in [KE03]. In our model, layers of more than one frame can be transmitted at each time slot which allows speeding up or slowing down frame scheduling according to the channel conditions and to the buffer state. Thus, a variable delay is obtained between the decision time and the displaying time. The delay is variable but not randomly as in [KE03] since it depends on the transmission action considered by the layer filtering process. The main difficulty comes from the presence of this delay, where the layer filtering process located at the transmitter side should accurately estimate the PSNR with which the transmitted frame will be displayed.

In summary, a video scheduling control system explicitly considering the delayed nature of the channel state as well as both the heterogeneous characteristics of the multimedia traffic and the time-varying wireless conditions is still missing.

1.2 Main contributions

In this chapter, we design a cross-layer control mechanisms cast in the MDP framework to control the level of buffers at MAC and at Application layers at the transmitter side *only* of a communication chain able to efficiently estimate the PSNR of the decoded frames.

Feedback at the MAC layer is implicitly used, but no feedback at the Application layer, which may include information about channel and receiver buffer states, is considered avoiding the use of delayed measurements. The control is performed by a filtering of scalability layers of the encoded video. A Markov model of the channel is considered using three hypotheses concerning the knowledge of the state of the channel, namely, known state, known state with delay, and unknown state. In the last two cases, the design of a control law may even be cast in a classical MDP framework. The problem of frame priority level is also addressed while choosing the scheduling decision among frames.

Our model is more general than that considered in [dCR03], because layers of more than one frame can be transmitted at each time slot which allows speeding up or slowing down video transmission according to the channel condition and to the buffer state. A challenging task in delivering multimedia data over wireless network is the dynamic characteristics of both the wireless channel and the multimedia source data. To overcome this problem, as in [FVdS10], we consider learning techniques

which allow an on-line update of the filtering decision according to the changing system characteristics. In this chapter, to limit the feedback messages from the application layer, we do not consider the receiver buffer state while various levels of knowledge of the state of the channel are considered. In addition, in order to perform a fast learning process, we need to reduce the system state space considered in the filtering process in Chapter 7. Thus, the L post-encoder buffers are represented by a single shared post-encoder buffer.

2 System description

In this section we consider a video scheduling system for mobile users. The considered single user unicast transmission system is sketched in Figure 8.1. The core network consists of a streaming server hosting a scalable video coder, a *proxy*, and a base station. Packets are transmitted through a wireless channel and received by a mobile client. Among the components of the base station (Node B), we consider mainly the MAC buffer. The MAC scheduler of Node B, as well as its physical layer, its radio front-end, the wireless channel, the physical layer of the receiver, and the part of the MAC layer at receiver side managing ACK/NACK procedures are denoted in the remainder of the chapter as the *channel*.

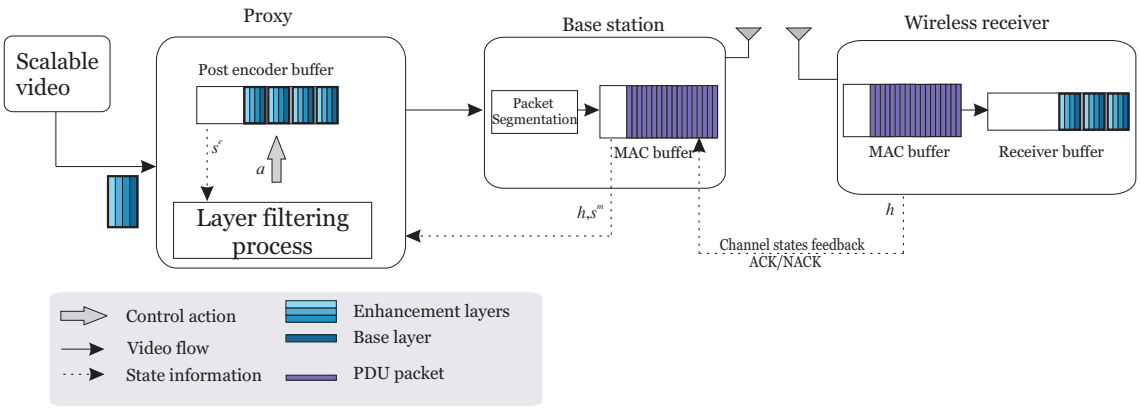


Figure 8.1: Wireless scalable video transmission system.

If some parts of the system are uncontrolled (the source, the channel), most other parts may be tuned to satisfy system requirements. Usually, one aims at maximizing the quality at receiver side, this quality should vary smoothly with time, transmission delays should be minimized, etc. The time-varying characteristics of the source and of the channel make the end-to-end optimization of such transmission chain quite difficult. Our goal is to design a quality aware adaptive and selective SNR layer transmission algorithm.

2.1 Streaming server

As in Chapter 7, the video sequence is segmented into frames and encoded into L layers: a base layer and a set of $L-1$ enhancement layers. Frames are generated with a constant period of time T and are identified by their temporal index t. The

encoding parameters (quantization step, frame rate, etc.) are controlled by the streaming server, independently of the remainder of the chain.

2.2 Proxy

The L SNR layers are packetized by the streaming server and fed, via an idealized network (lossless), to the post-encoder buffer. The proxy is assumed not to control the encoding parameters. At each time, the controller performs *layer filtering* within the proxy so that for each frame, SNR layers may be sent, kept, or dropped. The layer filtering process, performed by the proxy, should be designed to maximize the video quality at receiver side. To perform this task, the decoded video quality is maximized by taking into account most factors impacting directly or indirectly the received video quality: frame type, number of SNR layers, error concealment, packet loss due to post-encoder and MAC buffer overflow.

In order to limit the feedback delay, the proxy is placed close to the base station. In fact, the layer filtering process may be done in the proxy or in the base station.

2.3 Base station and channel

The base station contains additional buffers for each user connected to it to perform the rate and bandwidth allocation (MAC scheduling, see [FL02, AH05]) among users.

Transmitted packets by the layer filtering are fed to the MAC buffer of the base station after being segmented into Packet Data Units (PDUs) of constant size. One has to control the MAC buffer in order to avoid overflow state to prevent PDUs from being dropped. PDUs are then transmitted to the mobile receiver via a wireless channel characterized by time-varying conditions.

If the channel state is considered in the filtering control, the control process should exploit some feedback from the MAC buffer and from the mobile client to estimate the channel conditions. Without channel state observation, the control has to rely on the observation of the level of the MAC buffer only.

2.4 Receiver

The mobile receiver stores correctly received PDUs in its own MAC buffer. Packet de-encapsulation and buffering in the buffer at application layer is done as soon as all corresponding PDUs have been received. Complete or incomplete frames are then processed by the video decoder. Outdated packets are dropped, without being decoded.

Some packet-loss concealment may be put at work at the receiver side. The handover issues is not addressed. This means that we assume that the seaming server transmit video to a mobile receiver considered linked to the same base station during the streaming session.

3 Model description

3.1 MDP formulation

In this section, the problem of designing an optimal scheduling policy of L SNR scalable layers over a wireless channel is translated in the framework of discrete-time MDP.

We formulate the stochastic optimization problem as an MDP and solve it on-line using reinforcement learning. The advantages of the on-line method are that it does not require a priori knowledge of the traffic arrival and channel statistics to determine the scheduling policies. To solve our proposed optimization problem, all components of the tuple $(\mathcal{S}, \mathcal{A}, P, r)$ have to be identified.

We assume that time is slotted into discrete-time intervals of length T such that the t-th time slot is defined as the time interval $[tT, (t+1)T)$. T can be set at the frame level corresponding to the cadence of the encoder or at the PDU level corresponding to cadence of the scheduler at the MAC layer. Filtering decisions are made at the beginning of each interval and the system state is assumed to be constant throughout each interval.

3.2 System states

The set of states of the systems are: h is the channel state, s^I is the frame type state, s^e is the post-encoding buffer state, and s^m is the MAC buffer state. The vector gathering all state is $\mathbf{s} = (h, s^I, s^e, s^m) \in \mathcal{S}$.

3.2.1 Channel state

The channel state h_t describes the channel conditions (rate, probability of error, capacity, *etc.*) assumed constant between t and $t+1$. The varying channel rate is modeled here as an N_h-state Markov chain as in [COR02] and [FVdS10]. At time t, the state h_t with values in the set $\mathcal{H} = \{1, \ldots, N_h\}$ represents a rate within the set $\mathcal{R}^c = \left\{R_0^c, \ldots, R_{N_h}^c\right\}$ expressed in bits/s.

State transition probability $p_{x,k} = p(s_t^h = x \,|\, s_{t-1}^h = k)$ from moving from state $x \in \{1, \ldots, N_h\}$ to state $k \in \{1, \ldots, N_h\}$ may be estimated on-line.

Three hypotheses concerning the knowledge of the state of the channel are considered.

- *Hyp. 1*: *instantaneously available channel state*, where h_t is assumed available when choosing the action to apply between time t and $t+1$; this is realistic only when feedback with very short delay is possible.
- *Hyp.2*: *Delayed channel state*, where the channel state is assumed available after a delay δ when choosing the action to apply between t and $t+1$; represents a more realistic situation.
- *Hyp.3*: *unknown channel state* which is a scenario, where no channel state feedback is considered.

3.2.2 Frame type state

The frame can be classified as being one of the possible types depending on the specific video coder being used at the video server. These types can be defined for example based on frame activity level as in [HCC08], type of the frame (I,P,B) as in [TC01], or size of the frame as in [MvdS11], etc. Video streams are typically compressed into GoP structures containing intra-predicted (I), inter-predicted (P), and bi-directionally predicted (B) data units (*e.g.*, MPEG-2, MPEG-4, and H.264/AVC). We consider frame types I, P, and B to illustrate the dependencies between frames as well as the impact of each frame type on the video quality. It has been shown that transitions among data unit types in a GoP structure can be modeled as a stationary Markov process [TC01]. We assume that the choice of frame type is set constant by the encoder for the whole video sequence; therefore, the frame type transition probabilities depend on the desired ratio of I, P, and B data units (*e.g.*, IBPBPBP ...) set by the video coder. $s^I \in \mathcal{S}^I$ denotes the frame state.

Let $b_t(s_t^I, \ell)$ be the encoded bit rate (bits per SNR layer) of SNR layer $\ell \in \{1, \ldots, L\}$ of a frame t and $p_t(s_t^I, \ell)$, represents the additional PSNR when receiving the $\ell - th$ layer of frame t. The bit rate and the PSNR value depend on the frame type.

We assume that $b_t(s_t^I, \ell)$ and $p_t(s_t^I, \ell)$ are instances of *i.i.d.* random variables $B(s_t^I, \ell)$ and $P(s_t^I, \ell)$ with distribution $p_B(b_t|s_t^I, \ell)$ and $p_P(p_t|s_t^I, \ell)$, respectively.

3.2.3 Buffer state

The state of the post-encoder buffer is denoted by $s^e \in \mathcal{S}^e$ and the state of the MAC buffer is denoted by $s^m \in \mathcal{S}^m$. Here, the state of the post-encoder buffer describes the number of frames stored in the buffer. This help to regulate the delay introduced within the system. The state of the MAC buffer corresponds to the number of PDUs or of bits (PDUs have all the same size) in the buffer.

3.3 Actions

The layer filtering process is in charge of deciding the number of SNR layers to send among the two oldest frames in the post-encoder buffer.
$\mathbf{a}_t = (a_{1,t}, \ldots, a_{L,t}, a_{L+1,t}, \ldots, a_{2L,t}) \in \mathcal{A}$ is the vector of actions taken in the post-encoder buffer between time t and $t+1$, where $a_{l,t} = \{-1, 0, 1\}$ for $\ell \in \{1, \ldots, 2L\}$ is the filtering decision for the $\ell - th$ SNR layer of the last two frames in the buffer. $a_{\ell,t}$ represents the number of transmitted packets from the post-encoder buffer, when its value is positive, or the number of dropped packets when it is negative. If $a_{\ell,t} = 0$, packets are neither transmitted nor dropped.

3.4 State transition probability matrix

Once all states and actions have been identified, one has to determine the transition between states as well as the state transition probability matrix

$$P\left(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t\right) = \Pr\left(\mathbf{s}_{t+1} \,|\, \mathbf{s}_t, \mathbf{a}_t\right), \tag{8.1}$$

with $\mathbf{s} = (h, s^I, s^e, s^m) \in \mathcal{S}$, assuming that the transitions from $\mathbf{s}_t$ to $\mathbf{s}_{t+1}$ only depends on the state $\mathbf{s}_t$ and action vector $\mathbf{a}_t$. Details on the state transitions probability derivation of the various system states involved in the filtering process are in Appendix C.

Now, the transition probability matrix may be evaluated in the three hypotheses concerning the knowledge of the state of the channel:

Hyp. 1: In this case, h_t is available at time t when applying $\mathbf{a}_t$ to the system. The state transition matrix is then

$$\begin{aligned} P_1(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t) &= Pr\left(s^e_{t+1}, s^m_{t+1}, s^I_{t+1}, h_{t+1} | s^e_t, s^m_t, s^I, h_t, \mathbf{a}_t\right) \\ &= p(s^e_{t+1}|s^e_t, \mathbf{a}_t) p(s^I_{t+1}|s^I_t, \mathbf{a}_t) p(s^m_{t+1}|s^m_t, h_t, s^I_t, \mathbf{a}_t) p(h_{t+1}|h_t). \end{aligned} \tag{8.2}$$

Hyp. 2: In this case, the current state of the channel is available without noise with a unit delay. In [AN92], the case $\delta = 1$ was considered and the delayed MDP model is transformed into a standard fully observable MDP. Here, the situation is simpler, and one may obtain a new state transition matrix using marginalization and the joint and conditional probability rule. We obtain

$$\begin{aligned} P_2(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t) &= Pr\left(s^e_{t+1}, s^m_{t+1}, s^I_{t+1}, h_t | s^e_t, s^m_t, s^I, h_{t-1}, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} Pr\left(s^e_{t+1}, s^m_{t+1}, s^I_{t+1}, h_{t+1}, h_t | s^e_t, s^m_t, s^I, h_{t-1}, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} Pr\left(s^e_{t+1}, s^m_{t+1}, s^I_{t+1}, h_{t+1} | s^e_t, s^m_t, s^I, , h_t, h_{t-1}, \mathbf{a}_t\right) Pr\left(h_t | s^e_t, s^m_t, s^I, h_{t-1}, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} P_1(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t) \Pr\left(h_t | s^e_t, s^m_t, s^I, h_{t-1}, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} P_1(\mathbf{s}_t, \mathbf{s}_{t+1}, \mathbf{a}_t)\, p(h_t | h_{t-1}). \end{aligned}$$

This result is obtained since h_t is known, and other states and $\mathbf{a}_t$ do not provide additional information on h_t. Thus P_2 may be easily derived from P_1.

Hyp. 3: In this case, no channel state is available to the controller. The state transition matrix may then be written as

$$\begin{aligned} P_3(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t) &= Pr\left(s^e_{t+1}, s^m_{t+1}, s^I_{t+1} | s^e_t, s^m_t, s^I, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} Pr\left(s^e_{t+1}, s^m_{t+1}, s^I_{t+1}, h_{t+1} | s^e_t, s^m_t, s^I, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} \sum_{h_t} Pr\left(s^e_{t+1}, s^m_{t+1}, s^I_{t+1}, h_{t+1}, h_t | s^e_t, s^m_t, s^I, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} \sum_{h_t} Pr\left(s^e_{t+1}, s^m_{t+1}, s^I_{t+1}, h_{t+1} | s^e_t, s^m_t, s^I, h_t, \mathbf{a}_t\right) Pr\left(h_t | s^e_t, s^m_t, s^I, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} \sum_{h_t} P_1(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t) \Pr\left(h_t |\, s^e_t, s^m_t, s^I, \mathbf{a}_t\right) \\ &= \textstyle\sum_{h_{t+1}} \sum_{h_t} P_1(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t)\, p(h_t), \end{aligned}$$

since other states or $\mathbf{a}_t$ do not provide additional information on h_t.

3.5 Reward function

The layer filtering process should be designed to maximize the video quality at the receiver side. To regulate the video quality, several objective and subjective video quality measurement techniques are available, see, *e.g.*, [ITU08, SW98, Win05] and the references therein. Here, as in most of existing works, we use the Peak-Signal-to-Noise Ratio (PSNR) metric. The PSNR of a frame is $P = 10 \log_{10}\left(\frac{255^2}{D}\right)$ where D is the distortion of the frame (considering a quadratic distortion measure). Other quality metric can be considered. To reach good visual video quality, one may also control the packet loss due to frame drop of buffer overflow.

One has to define the reward function that the optimization process should maximize in order to maximize the PSNR of the decoded frame. Ideally, one should estimate and maximize the PSNR of the decoded frame. However, it is difficult to efficiently estimating it at the controller due to the delay caused by the MAC and the receiver buffers.

To tackle the problem of transmission delay, the proposed layer filtering process should be able to estimate the decoded PSNR at each time based on the chosen action, the environment condition, and the actual buffer state. Thus, the reward function to be maximized by the MDP framework a PSNR estimated at the transmitter.

We define the reward function to be maximized by the MDP framework. The reward function is calculated by taking into account most factors impacting directly or indirectly the received video quality. These factors are

- number of SNR layers,
- frame type (I or P frame),
- error concealment (impact of frame drop),
- packet lost due to post-encoding and MAC buffer overflow.

When the proxy decides to transmit $k \leq L$ layers of the last frame in the post-encoder buffer, the estimated PSNR should be derived from the chosen action $\mathbf{a}_t$, the PSNR value in each SNR layer of the concerned frame, and also from the probability of being dropped from the buffer. This probability is derived from the post-encoder and the MAC buffer states which will be used to know whether there is or is not enough space in these buffers before transmitting or keeping the frame.

Thus, we propose to decompose the PSNR estimation into two steps. In the first step, the estimator calculates the PSNR of the transmitted frame regardless of the MAC buffer state. In the second step, the estimator calculates the next state of the MAC buffer using the chosen action in order to estimate whether the frame will be safely stored of dropped from the buffer. We assume that the receiver buffer is filled in a progressive fashion which allows the viewer to decode as new frame arrives in the buffer.

In the first step, we distinguish two cases: SNR layers of only the last frame in the post-encoder buffer are transmitted, and SNR layers of both the last and the last but one frames in the post-encoder buffer are transmitted. The PSNR value $p(s_t^I, \ell)$ is assumed to be a realization of the i.i.d. random variable $P(s_t^I, l)$ with distribution $p_P(p_t|s_t^I, \ell)$. The PSNR reward function is then calculated as follow

$$r_t^q\left(s_t^I, \mathbf{a}_t\right) = \sum_{k=1}^{L} \max(0, a_t(2L+1-k))p_t(s_t^I, k), \tag{8.3}$$

corresponding to the summation of PSNR contributions of each transmitted layer of the last frame in the post-encoder buffer.

In the second case, the estimation is less trivial than in the first case since two frames are transmitted simultaneously. The reward should contain PSNR of the first and of the second transmitted frames. A simple solution would be to calculate the average PSNR of the two transmitted frames

$$\begin{aligned} r_t^q\left(s_t^I, \mathbf{a}_t\right) = \; & \tfrac{1}{2}\left(\sum_{k=1}^{L} \max(0, a_t(2L+1-k))p_t(s_t^I, k)\right) \\ & + \tfrac{1}{2}\left(\sum_{k=1}^{L} \max(0, a_t(L+1-k))p_{t+1}(s_t^I, k)\right), \end{aligned} \tag{8.4}$$

corresponding to the summation of PSNR contribution of each transmitted layer of the two last frames in the post-encoder buffer. The reward functions in (8.3) and (8.4) does not consider the frame drop action. In addition, frame drop can be caused by the post-encoder buffer or the MAC buffer overflow. In order to tackle the problem of packet loss in the post-encoder buffer and the MAC buffer due to the overflow situation, we assume that when the post-encoder buffer is full, frames are dropped in a head of line order (the oldest frame in the buffer is dropped first), however, when the MAC buffer is full frames are dropped not in a head of line order but the last introduced frame in the MAC buffer causing overflow will be dropped so that the layer filtering process could estimate according to the buffer state the drop probability. Let P_{d} be the probability that the transmitted frame(s) will be dropped. This frame can be dropped only if the layer filtering process decides to drop the frame, or if the layer filtering process decides to keep the frame in the post-encoder buffer which is already full, or if the layer filtering decide to transmit the frame but the MAC buffer has no more space to store it. Let P_{e} is the probability that the frame is dropped because the post-encoder buffer is in an overflow situation, P_{m} is the probability that the frame is dropped because the MAC buffer is in an overflow situation, and P_{a} is the probability that the layer filtering process decides to drop the frame. The last probability is $P_{\mathrm{a}}(a_t) = 1$ when a_t is a drop action and $P_{\mathrm{a}}(a_t) = 0$ otherwise. We notice that if a_t is a drop action the post-encoder buffer will be able to store the next frame transmitted by the video encoder and so $P_{\mathrm{e}} = 0$, moreover, no additional PDU packets will be stored in the MAC buffer and so $P_{\mathrm{m}} = 0$. Otherwise, if the layer filtering decides to transmit one or two frames then $P_{\mathrm{m}} \in \{0, 1\}$ and $P_{\mathrm{e}}(s_t^e, a_t) = P_{\mathrm{a}}(a_t) = 0$. Thus, the probability P_{d} that the transmitted frame(s) will be dropped can be expressed as

$$P_{\mathrm{d}}(\mathbf{s}_t, a_t) = P_{\mathrm{e}}(s_t^e, a_t) + P_{\mathrm{m}}(s_t^m, a_t) + P_{\mathrm{a}}(a_t), \tag{8.5}$$

In case of non reception of the base layer of the reference frame, error concealment techniques, in order to reconstruct the missing frame, are used. Various algorithms exist for frame loss/ error concealment [WB06]: *frame copy* algorithm, *motion vector copy* algorithm, etc. If the *Frame copy* algorithm is performed when a frame is lost, the PSNR is assumed to be reduced by $\lambda(s_t^I)$ due to the loss of the current frame and depending on the frame type. These values can be estimated experimentally off-line for some tunning video sequences.

The total reward function is

$$r_t\left(\mathbf{s}_t, \mathbf{a}_t\right) = \left(1 - P_{\mathrm{d}}(\mathbf{s}_t, a_t)\right) \cdot r_t^q\left(s_t^I, \mathbf{a}_t\right) + P_{\mathrm{d}}(\mathbf{s}_t, a_t) \cdot \left(r_{t-1}\left(\mathbf{s}_{t-1}, \mathbf{a}_{t-1}\right) - \lambda(s_t^I)\right). \tag{8.6}$$

In order to learn the optimal action to choose at each state, an on-line learning is used. More details about reward update are provided in the next section.

3.6 Learning the optimal policy

In practice, the reward and transition probability functions are (partially) unknown a priori. Consequently, the optimal state-value function V^* and the optimal policy π^* cannot be computed using value iteration. Thus, we adopt a model-free reinforcement learning approach.

We define the optimal action-value function $Q^* : \mathcal{S} \times \mathcal{A} \mapsto \Re$, which satisfies:

$$Q^*(s_t, a_t) = r(s_t, a_t) + \gamma \sum_{s_{t+1} \in \mathcal{S}} P(s_{t+1}, s_t, a_t) V^*(s_{t+1}), \tag{8.7}$$

where $V^*(s_{t+1}) = \max_{\mathbf{a} \in \mathcal{A}}(Q^*(s_{t+1}, a)$. Learning is performed using the Q-learning algorithm described in Chapter 6. Q-learning is an RL technique that works by learning an action-value function that gives the expected reward of taking a given action in a given state and following a fixed policy thereafter. The action-value-function is updated at each time slot according to

$$Q_{t+1}(s_{t+1}, a_{t+1}) \leftarrow (1 - \alpha_t) Q_t(s_t, a_t) + \alpha_t [r_t + \gamma \max_{a \in \mathcal{A}} Q_t(s_{t+1}, a'_t)] \tag{8.8}$$

where a'_t is the greedy action in state s_{t+1}, which maximizes the current estimate of the action-value function; $\alpha \in [0, 1]$ is a time-varying learning rate parameter; and, $Q_0(s, a)$ can be initialized arbitrarily for all $(s, a) \in \mathcal{S} \times \mathcal{A}$. Here, $Q_0(s, a)$ is initialized to zero for each state action pair.

At the beginning of the control process, the controller should go through the same environment many times in order to learn how to find the optimal actions. During the exploration step, the Q-learning rule in (8.8) is performed by executing each action in each state a number of times until the Q values converges to the optimal value.

In order to make the controller aware about the different changes that may happen in the system (channel condition, video characteristics, etc.), we perform ϵ-greedy where the controller identifies the best action according to the state-action values but also with probability ϵ rather than taking the best action, the controller will uniformly select an action from the remaining actions and update the state-action values accordingly. An other strategy for dealing with large state spaces is to treat them as a hierarchy of learning problems [BM03]. Hierarchical solutions allow reducing the execution time, learning time, and space. We use the *virtual experience* method proposed in [MvdS11] to update the state-action value.

In fact, the large number of states significantly limits the system learning speed because the Q-learning update step defined in (8.8) updates at time t the action value function $Q_t(s_t, a_t)$ for only one state-action pair (s_t, a_t). The virtual experience algorithm exploits the form of the transition probability and reward functions in order to update the action value function for multiple statistically equivalent state-action pairs (s, a_t) in each time slot, including those that have never been visited.

Figure 8.2(a) illustrates how Q-learning updates a state-action pair (s_t, a_t). Figure 8.2(b) illustrates how performing Q-learning update on every virtual experience tuple $(s(k), a_t)$, for $k \in \{1, \ldots 5\}$, where five corresponds to the number of system state considered in the example in Figure 8.2.

In our specific setting, we exploit the fact that the data unit arrival distribution is independent of the current buffer state and of the channel state. This allows us to extrapolate the experience obtained in each time slot to other buffer states and action pairs. Thus, in stationary environments, the virtual experience algorithm improves convergence time at the expense of increased computational complexity. More details about the method are available in [MvdS11].

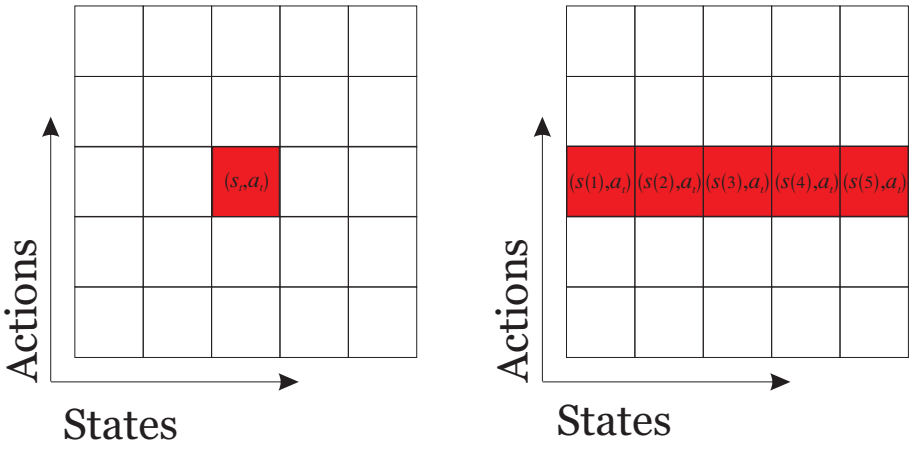


Figure 8.2: State-action pairs that are impacted in one time slot when using different learning updates (highlighted in red). (a) Q-learning update for state-action pair (s_t, a_t) pair ; (b) Virtual experience updates for (s, a_t) pairs.

4 Experimental results

4.1 Simulation conditions

The performance of the proposed layer filtering process has been evaluated on *Foreman.qcif* and *Mother & Daughter.qcif* sequences at $F = 30$ fps are reported. Experiments are performed using the H.264/SVC encoder (in version JSVM 9.19) [VWS07]. The temporal period at which the control system is operating is taken as $T = 1/F$.

In this section, we evaluate the performance of our layer filtering process using the three hypotheses concerning the knowledge of the state of the channel. Our system is compared to a basic content independent bit stream extractor described later.

4.1.1 Video encoder

The video sequences are divided into GoPs of $N_G = 16$ frames. Each GoP is encoded with two temporal scalability levels indicated by the temporal identifier T_0 and T_1. The first frame in the GoP is encoded in Intra mode and the remaining frames are encoded as P-frames such that the I frame is identified by T_0 and the P frames are identified by T_1.

Video sequences are encoded using three MGS scalability layers per frame ($L = 3$) corresponding to different video qualities. Each MGS scalability layer is identified by a quality identifier Q (Q_0, Q_1, Q_2). Thus, each frame in the sequence is identified by the pair (T,Q). Tables 8.1 and 8.2 reports the characteristics of the three MGS layers in terms of bit rate (expressed in kbit/s) and associated Peak Signal-to-Noise Ratio (PSNR) of the luminance component (Y PSNR) (expressed in dB).

In the proposed filtering process, the number of transmitted enhancement layers is adjusted for each frame mainly according to the channel constraint and to the buffers occupancy levels. Dropping some of the enhancement layers may introduce some drift. This effect is enhanced when the reference frames taken for motion prediction are those with the highest available quality. A way to minimize the drift effect is to consider all frame encoded as key pictures in Medium-grain quality

Quality Id	QP	bit rate (kbit/s)	Y PSNR (dB)
Q 0	40	39.97	28.36
Q 1	34	244.71	32.10
Q 2	29	382.55	35.73

Table 8.1: Encoding details of Foreman sequence

Quality Id	QP	bit rate (kbit/s)	Y PSNR (dB)
Q 0	40	14.61	30.49
Q 1	34	142.05	33.50
Q 2	29	263.16	37.12

Table 8.2: Encoding details of Mother and daughter sequence

scalable (MGS) scheme [SMW07]. Thus, with this configuration only the base layer is considered in the inter-frame prediction. The price to be paid is a reduced coding efficiency.

In the considered transmission system, frames may be lost when the layer filtering process chooses to drop it or due to buffer overflow. In this theses, we consider a frame copy algorithm, *i.e.*, if the current frame is lost, the previously decoded frame is copied to the current frame position.

4.1.2 Channel

A time-varying channel is considered in the simulations. It consists of a two-state Markov model, which state switches between *bad* (B) state with instantaneous channel rate R_0^c and *good* (G) state with instantaneous channel rate R_1^c within a period T. The channel state transition are governed by the following transition probabilities $P(G|G) = 0.9$ and $P(B|B) = 0.8$ and stationary probabilities $P(G) = 0.66$ and $P(B) = 0.33$. A quite large value of $P(B|B)$ has been chosen to simulate the bursty nature of an error-prone wireless channel.

4.1.3 Buffers

The post-encoder buffer and the MAC buffer are assumed having respectively a maximum size B^e in term of number of frame and B^m in term of number of PDU packets. As specified in the Radio Link Control (RLC) protocol specification of the the 3rd Generation Partnership Project (3GPP), the PDU size can be semi static [V5.05], *e.g.*, 336 bits or 656 bits, or flexible [V7.09] as introduced in Release 7 in order to decrease RLC protocol overhead and to avoid sequence number stalling. In our simulations, we consider the PDU to be static with a PDU size equal to 336 bits. The level of the post-encoder buffer has to be controlled to avoid frame drop due to buffer overflow and to limit the delay introduced by the system.

In order to reduce the model state space and accelerate the convergence of the RL process, we quantify the post-encoder buffer states in number of frames into 2 intervals. $\mathcal{S}^e = \{1, 2\}$, where $s^e = 1$ indicates a satisfying occupancy, with an actual number of frame between $b^e(1) = 0$ and $b^e(2)$. The state $s^e = 2$ corresponds to an almost full state when its actual level is larger than $b^e(2)$ but less than $b^e(3) = B^e$. Concerning the MAC buffer, finer granularity than that used in the post-encoder

buffer is used since the MAC buffer state in PDU transition depends on the frame size contrary to post-encoder buffer state. In addition, when the channel state is not available, the control has to rely on the observation of the level of the MAC buffer only. MAC buffer states are quantized into five intervals. $\mathcal{S}^m = \{0, 1, \ldots, 5\}$ with $s^m = j$, $j \in \{1, \ldots, 5\}$ corresponds to a PDU numbers between $b^m(j)$ and $b^m(j+1)$ with $b^m(1) = 0$ and $b^m(6) = B^m$. The fifth interval is considered smaller than the other intervals in order to avoid MAC buffer overflow and prevent PDUs from being dropped. The post-encoder and MAC buffers bounds considered in our experimental test are reported in Table 8.3.

$b^e(1)$	$b^e(2)$	$b^e(3)$	$b^m(1)$	$b^m(2)$	$b^m(3)$	$b^m(4)$	$b^m(5)$	$b^m(6)$
0	23	25	0	120	240	360	480	500

Table 8.3: Post-encoder and MAC buffer state parameters.

In order to perform virtual experience, knowledge about the buffer state transition probabilities is required. We assume states to be uniformly distributed within the different intervals. Transitions probabilities may be evaluated using (8.2). Some buffers states transition probabilities may depends on the system dynamic. These probabilities are estimated on-line.

4.1.4 Rewards

The layer filtering process in the proxy has to determine the number of SNR layer to transmit from the two last frames in the post-encoder buffer such that the cumulated discounted reward is maximized.

The decision is represented by set of actions $\mathbf{a}_t = (a_{1,t}, \ldots, a_{3,t}, a_{4,t}, \ldots, a_{6,t}) \in \mathcal{A}$ is the vector of actions taken in the post-encoder buffer between time t and $t+1$, where $a_{\ell,t} = \{-1, 0, 1\}$, $\ell \in \{1, \ldots, 6\}$ corresponding to the scheduling decision for the l-th SNR layer of the two oldest frame in the post-encoder buffer. Actions of the post-encoder buffer considered in our system are listed in Tab. 4.1.4.

The video quality reward function is defined as in (8.6). The value of the PSNR reduction $\lambda(s_t^I)$ is a predefined value that depends on the frame type. $\gamma(1) = 15$ dB

Action index	Action
1	$\mathbf{a} = (0, 0, 0, 0, 0, 0)$
2	$\mathbf{a} = (0, 0, 0, -1, -1, 1)$
3	$\mathbf{a} = (0, 0, 0, -1, 1, 1)$
4	$\mathbf{a} = (0, 0, 0, 1, 1, 1)$
5	$\mathbf{a} = (-1, -1, 1, 1, 1, 1)$
6	$\mathbf{a} = (-1, 1, 1, 1, 1, 1)$
7	$\mathbf{a} = (1, 1, 1, 1, 1, 1)$
8	$\mathbf{a} = (-1, -1, 1, -1, -1, 1)$
9	$\mathbf{a} = (-1, -1, 1, -1, 1, 1)$
10	$\mathbf{a} = (-1, 1, 1, -1, 1, 1)$
11	$\mathbf{a} = (0, 0, 0, -1, -1, -1)$

Table 8.4: Considered actions for $L = 3$.

corresponding to the PSNR reduction when an I frame is lost and $\lambda(2) = 8$ dB when a P frame is lost for both Foreman and Mother & Daughter sequences.

4.2 Learning process

On-line learning is performed using the Q-learning technique over 10000 time slots (Foreman and Mother & Daughter sequences by repeating the sequences from the beginning after 300 frames) with a virtual experience learning update at each time slot. Q-learning algorithms works by estimating the values $Q(s, a)$ of state-action pairs (s, a). The reward function r_t corresponds to the estimated PSNR defined in (8.6). Once these values have been learned, the optimal action from any state is the one with the highest Q-value.

4.3 Results: System performance with immediately known, delayed and unknown channel state

The proposed stochastic scalable layer filtering process is implemented for the three levels of knowledge of the channel states: the channel state is immediately available when choosing an action, the channel state is available with a unit delay, and no channel state is available to the controller. These three cases are performed using myopic and foresighted policies and are compared to the Basic Bit-Stream Extraction (BBSE) in the same three mentioned cases.

4.3.1 Basic bit stream extractor

A BBSE method is a content independent extraction method provided in JSVM (Joint Scalable Video Model). The BBSE process consists in extracting SVC layer according to a specific priority order. The prioritization is done according to the high-level syntax elements dependency id: dependency_id, temporal_id and quality_id. Figure 8.3 illustrates the prioritization order used in the BBSE process. In Figure 8.3, each block represents a layer referred to as $L(D_d, T_t, Q_q)$ where D_d indicates the spatial resolution, T_t the temporal resolution and Q_q the quality level.

The base quality denoted by Q_0 of each spatial and temporal resolutions lower or equal to the target spatial and temporal resolutions appears first. Next, for lower spatial resolution, NAL units of higher quality level are ordered in increasing order of their temporal level. Finally, for the target spatial resolution, NAL units are ordered based on their quality level and are included until the target bit rate is reached. Here we only focus on the temporal and the quality levels. The number of SVC layers allowed to be transmitted for each frame are selected according to the defined prioritization order and to the MAC buffer state. The considered scalability consider two temporal scalability levels and three quality scalability levels. The priority order of the considered scalability scheme using the BBSE is in Figure 8.4. For the target temporal resolution, NAL units are ordered based on their quality level and are included in the MAC buffer in order to fill the most the empty space in the buffer without reaching the overflow state.

In order to compare the proposed scheduling system to the BBSE, we consider various levels of knowledge of the channel state in the BBSE system: instantaneous channel state, delayed channel state, and unknown channel state. The performance

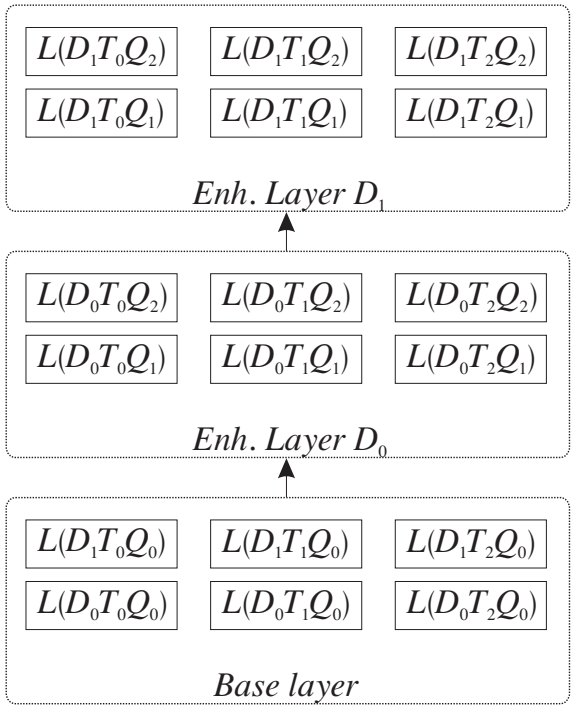


Figure 8.3: Priority order for NAL units using the basic bitstream extractor.

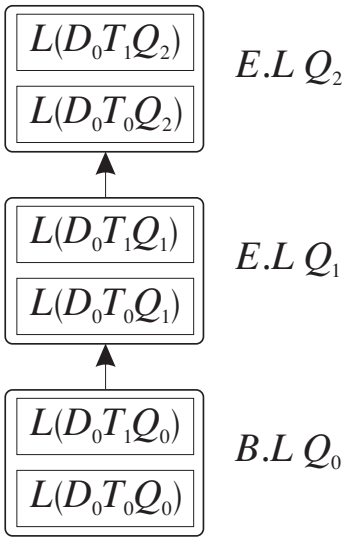


Figure 8.4: Priority order of the considered scalability scheme using the basic bitstream extractor

of the BBSE in the three cases are reported later. A major drawback of the BBSE method is that its prioritization policy is independent of the video content.

4.3.2 Results

Figures 8.5 and 8.6 show the performance of the system under the three hypothesis concerning the channel state knowledge in the proposed scalable layer filtering process for Foreman and Mother & Daughter sequences using the foresighted policy and compared to the BBSE process. Different channel rates are considered $[50-500]$ kbit/s for Foreman sequence and $[40-220]$ kbit/s for Mother and Daughter sequence.

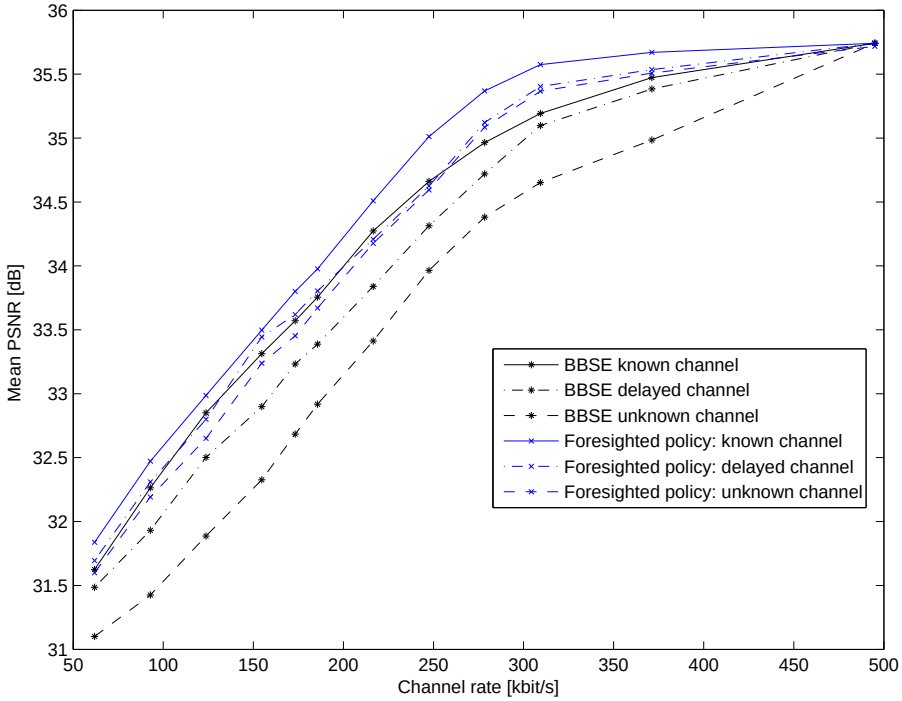


Figure 8.5: PSNR of the decoded Foreman sequence under the three hypothesis with the foresighted policy.

As demonstrated in Figures 8.5 and 8.6, when the channel state is considered known at each regulation time, the proposed foresighted scheme outperforms the BBSE scheme by a maximum of over 0.4 dB and an average of 0.23 dB for the Foreman sequence and over 0.39 dB and an average of 0.14 dB for the Mother & Daughter sequence. When the channel state information is considered delayed with a unit delay, our proposed foresighted scheme outperforms the BBSE scheme by a maximum of over 0.54 dB and an average of 0.32 dB for the Foreman sequence and over 0.43 dB and an average of 0.27 dB for the Mother & Daughter sequence. When the channel state information is not considered in the system model, our proposed foresighted scheme outperforms the BBSE scheme by a maximum of over 0.91 dB and an average of 0.64 dB for the Foreman sequence and over 0.84 dB and an average of 0.56 dB for the Mother & Daughter sequence.

The provided gain of the proposed scheme compared to the BBSE scheme is mainly due to the accurate SNR layers selection choice which relies on the contri-

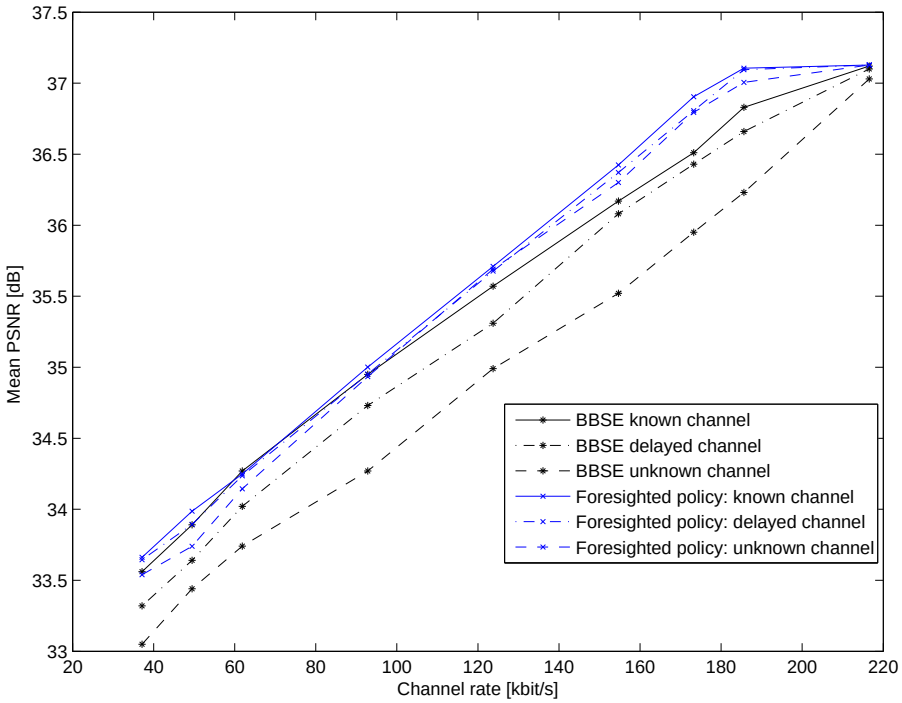


Figure 8.6: PSNR of the decoded Mother & Daughter sequence under the three hypothesis with the foresighted policy.

bution of each layer to the video quality, contrary to the BBSE scheme which only uses the high level syntax elements of the NAL units to order the SNR layers, and, thus, it is unaware of the impact of each layer on the quality of the sequence.

In our simulation, we considered a two-state Markov model for the channel state. This assumption can limit the performance of the proposed filtering system. Higher gain may be achieved while consider less coarse channel granularity model. Thus, we plan to implement the proposed layer filtering process in a prototype LTE network that includes a completely functional eNB and user equipment in order to assess its performance. Many elements should be considered due to the versatility of the LTE network such as MAC and RLC buffers management, terminal mobility, channel variability, video characteristics. This will allow us to measure the reliability of the considered channel model.

5 Conclusion

This chapter proposes an on-line solution for scalable video transmission over a time-varying wireless channel. A challenging task in delivering multimedia data over wireless network is the dynamic characteristics of the wireless channel. To overcome this problem, we consider learning techniques which allow an on-line update of the scheduling decision according to the dynamic change of the environment. Moreover, in order to tackle the problem of delayed feedback of the channel state, we propose an on-line scalable packet scheduling and buffer management in both Application and Medium Access Control (MAC) layers where feedback at MAC layer is implicitly used, but no feedback at Application layer is considered, avoiding the use of delayed

measurements. Various levels of knowledge of the state of the channel are considered. Foresighted policies are developed using three models of the system characterized by an immediately available, a delayed, an unavailable channel state at the controller. The level of post-encoder buffers and that of the MAC buffer of the base station are controlled. Experimental results show that with delayed channel state information or without channel state information, the performance of the proposed foresighted control system is only slightly degraded compared to a case where the channel state information is available. The performance degradation is larger using the basic bitstream extractor scheme. The observation of the level of the MAC buffer provides a satisfying estimate of the channel state. This result is useful, since using channel state in the control system requires processing feedback messages, which provide a delayed estimate of the channel state. Nevertheless, higher gain may be obtained while considering less coarse channel granularity model. Implementing the proposed layer filtering process in a prototype LTE network is planned in order to assess the system performance and measure the reliability of the considered channel model.

Chapter 9

Conclusions and perspectives

In this dissertation, we developed control systems for the transmission of single and multiple video streams over time-varying channels. The quality of service requirements considered in our control systems are mainly focused on the efficient use of the available bandwidth, on video quality, and on limited transmission delay.

First, in order to transmit several video program in parallel over a shared broadcast channel while ensuring an efficient use of the available bandwidth, a smooth video quality variation, a fair video quality among the transmitted programs, and a limited transmission delay, a statistical multiplexing (SM) system is proposed.

To satisfy the quality constraints, a predictive control taking into account the impact of the chosen encoding parameter in a video unit (VU) on the rate and the distortion (R-D) of futures VUs is performed.

In addition, the transmission delay is adjusted using a PID controller. The control process is performed in a closed loop. Each Buffer sends its buffering delay to the encoders control which use it to update the encoding rate constraint so that buffering delays in all buffers of the multiplexed program are equal to a reference delay.

Owing to the closed-loop control, similar buffering delay for all programs and an efficient use of the available channel rate are obtained.

The proposed SM system is performed in a centralized way where the controller and the video servers are considered located in the same place. In order to address the problem of spatially distributed video servers, we proposed a partly distributed SM system able to satisfy some video quality fairness constraint among programs without requiring exchange of information between servers.

The SM system is performed through two control process. The first control is centralized and done within the media aware network element (MANE) taking into account the quality fairness constraint. Then, the MANE feeds back to each video server the level of its associated buffer. Thus, the second control is distributed where each video server adjusts its encoding rate independently from the others by adapting its rate-distortion trade-off so that its buffer reaches some reference level. The SM problem is modeled as a feedback control system using PI controllers for both the bandwidth and the encoding rate control. Guidelines are provided for choosing parameters for the proposed controllers and the stability region of the system is characterized.

Second, we developed a cross-layer optimization of scalable video transmission over wireless network. A layer filtering algorithm designed at the network using

stochastic optimization tools is proposed. Here again, the system requirements are mainly focused on the efficient use of the available bandwidth, good video quality at the reception, and buffer management.

The proposed layer filtering process is formulated using an MDP framework that explicitly considers the cooperation of the Application layer and the MAC layer, the heterogeneity of the video data and the varying network conditions across time. In order to address the problem of delayed feedbacks between the receiver and the control system used to follow up the channel state evolution, various levels of knowledge of the channel state are considered, namely, immediately known state, known state with delay, and unknown state. The filtering process under the three hypothesis concerning the knowledge of the channel state are formulated in a classical MDP framework. Without channel state observation, the control can use the observation of the level of the MAC buffer only to efficiently estimate the channel state. A challenging task in delivering multimedia data over time varying channel is the dynamic change of the characteristics of the wireless channel. To overcome this problem, we consider learning technique which allows an on-line update of the filtering decision according to the changing characteristics.

The proposed SM system can be extended in two directions. First, the control process can be performed at finer granularity VU level. This adaptation requires a special focus on the dependencies between VUs and their different distortion impacts. Designing an SM system for interdependent VUs requires the use of dependent R-D models, such as those described in [LO98], [MVHB04b], or the IRS-AR model proposed in Chapter 3. Such models take into account the impact of the encoding parameters in the reference VU on the rate and the distortion of its corresponding predicted VU. The difficulty with these models is the increased modeling complexity compared to independent models. Thus, solutions for reducing dependent model estimation complexity are required. Second, the rate control process used in the proposed SM system can be performed using scalable video sources. This can be done by adjusting the number of scalable layers. The main difficulty is that not all rate points may be reached in a continuous way (this would require very fine granularity). In addition, the proposed algorithms can be implemented in real networks and systems to compare them to existing rate control and resource management algorithms and to gain understanding of implementation issues. A typical scenario for an SM application is the Mobile TV service delivery over evolved MBMS standard [ETS05].

The proposed on-line learning for scalable video transmission optimization can be extended also in several directions. First, there is potential to advance reinforcement learning theory in delayed environment by gaining a better theoretical understanding of the dynamic behavior of the proposed learning algorithms. Scenarios involving delayed feedback have generated interest within the academic community, leading to the development of several important theoretical results. Practical solutions for dealing with constant and variable observation and reward delays are still missing.

Second, the layer filtering process is designed for scalable stream where drift propagation effect is bypassed. The drift take effect when scalable layers are dropped which may cause quality degradation. In future work, we should model the drift effect and take it into consideration while layer filtering. In addition, the channel state is modeled as a two-states Markov model where each state corresponds to a channel rate. However, the channel state is considered at the frame level which duration

is much larger than the coherence time corresponding to the channel variation in wireless communications systems. Thus, we plan to implement the proposed layer filtering process in a prototype LTE network that includes a completely functional eNB and user equipment in order to assess its performance. Many elements should be considered due to the versatility of the LTE network such as MAC and RLC buffers management, terminal mobility, channel variability, video characteristics. This will allow us to measure the reliability of the considered channel model.

Then, the receiver is assumed to be linked to the same base station during the streaming session. The handover issue should be addressed in our future work where a layer filtering process should be able to interact with both the video mobility management and the transport optimization components.

References

[AFK+11] N. Amram, B. Fu, G. Kunzmann, T. Melia, D. Munaretto, S. Randriamasy, B. Sayadi, J. Widmer, and M. Zorzi. QoE-based transport optimization for video delivery over next generation cellular networks. *IEEE Symposium on Computers and Communications*, pp. 19 – 24, Kerkyra, October 2011.

[AH05] W. Ajib and D. Haccoun. An overview of scheduling algorithms in MIMO-based fourth-generation wireless systems. *IEEE Network*, 19(5):43 – 48, September 2005.

[AN92] E. Altman and P. Nain. Closed-loop control with delayed information. *ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems*, pp. 193 – 204, New York, June 1992.

[BBPR98] M. Barni, F. Bartolini, A. Piva, and F. Rigacci. Statistical modelling of full frame DCT coefficients. *In Proc. EUSIPCO*, pp. 1513 – 1516, Patras, September 1998.

[BHKH04] N. Baldo, U. Horn, M. Kampmann, and F. Hartung. RTCP feedback based transmission rate control for 3G wireless multimedia streaming. *PIMRC*, vol. 3, pp. 1817 – 1821, Barcelona, September 2004.

[BLL00] J. Bai, Q. Liao, and X. Lin. Hybrid models of the rate-distortion characteristics for MPEG video coding. *International Conference on Communication Technology Proceedings*, vol. 1, pp. 363 – 366, Beijing, August 2000.

[BM03] A. Barto and S. Mahadevan. Recent advances in hierarchical reinforcement learning. *Discrete-Event Dynamic Systems: Theory and Applications*, 13(4):41 – 77, January 2003.

[BNW99] L. Böröczky, A. Y. Ngai, and E. F. Westermann. Statistical multiplexing using MPEG-2 video encoders. *IBM Journal of Research and Development*, 43(4):511 – 520, July 1999.

[BP66] L. Baum and T. Pétrie. Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics*, 37(6):1554 – 1563, December 1966.

[BW99] J. L. Bander and C. C . White. Markov decision processes with noisecorrupted and delayed state observations. *Journal of the Operational Research Society*, 50(6):660 – 668, June 1999.

[CGJ06] Q. Chengsheng, C. Guobin, and L. Jilin. An efficient two-pass VBR encoding algorithm for H.264. *Int. Conference on Communications Circuits and Systems Proc.*, vol. 1, pp. 118 – 122, Guilin Guangxi, June 2006.

[Cha02] K. Chandra. *The Wiley Encyclopedia of Telecommunications*, chapter Statistical Multiplexing. Wiley, 2002.

[CHC07] C-H. Chu, H-P. Hung, and M-S. Chen. Variant bandwidth channel allocation in the data broadcasting environment. *International Conference on Mobile Data Management*, pp. 94 –101, Mannheim, May 2007.

[CHJBdV11] C. Chen, R.W Heath Jr., A.C Bovik, and G. de Veciana. Adaptive Policies for Real-Time Video Transmission: A Markov Decision Process Framework. *IEEE international Conference on Image Processing*, Brussels, September 2011.

[Cis09] Cisco visual networking index: Forecast and methodology, 2008-2013. Technical report, Cisco, 2009.

[CLS08] N. Changuel, Y. Leprovost, and B. Sayadi. *Scalable Video Coding: principles and benefits*. Alcatel Lucent white paper, vx/netn/t/08/0093 edition, July 2008.

[CMvdS+11] N. Changuel, N. Mastronarde, M. van der Schaar, B. Sayadi, and M. Kieffer. Adaptive scalable layer filtering process for video scheduling over wireless networks based on MAC buffer management. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 2352 – 2355, Prague, May 2011.

[COR02] J. Cabrera, A. Ortega, and J.I. Ronda. Stochastic rate-control of video coders for wireless channels. *IEEE transaction on circuits and systems for video technology*, 12(6):496 – 510, June 2002.

[CSK09a] N. Changuel, B. Sayadi, and M. Kieffer. Joint encoder and buffer regulation for statistical multiplexing of multimedia contents. *European Patent*, no. 9290584, Alcatel-lucent, July 2009.

[CSK09b] N. Changuel, B. Sayadi, and M. Kieffer. Method for allocating bit rates to video programs transmitted within a fixed rate channel. *European Patent*, no. 9305287, Alcatel-Lucent, April 2009.

[CSK09c] N. Changuel, B. Sayadi, and M. Kieffer. Predictive control for efficient statistical multiplexing of digital video programs. *Int. Packet Video Workshop*, pp. 1 – 9, Seattle, May 2009.

[CSK09d] N. Changuel, B. Sayadi, and M. Kieffer. Statistical multiplexing of video programs. *IEEE Vehicular Technology Magazine*, 4(3):62 – 68, September 2009.

[CSK10a] N. Changuel, B. Sayadi, and M. Kieffer. H.264/AVC inter frame rate distortion dependency analysis based on independent regime switching AR models. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 914 – 917, Dallas, March 2010.

[CSK10b] N. Changuel, B. Sayadi, and M. Kieffer. Joint encoder and buffer control for statistical multiplexing of multimedia contents. *IEEE Global Telecommunications Conference GLOBECOM*, pp. 1 – 6, Miami, December 2010.

[CSK10c] N. Changuel, B. Sayadi, and M. Kieffer. Scheduling method for data streaming in a telecommunication network infrastructure. *European Patent*, no. 808083, Alcatel-Lucent, September 2010.

[CSK10d] N. Changuel, B. Sayadi, and M. Kieffer. Video packet filtering algorithm based on only mac buffer management. *European Patent*, no. 808627, Alatel-Lucent, November 2010.

[CSK11a] N. Changuel, B. Sayadi, and M. Kieffer. Statistical multiplexing of distributed video streams. *European Patent*, no. 809506, Alcatel-Lucent, May 2011.

[CSK11b] N. Changuel, B. Sayadi, and M. Kieffer. Statistical multiplexing of distributed video streams. *Proceeding Gretsi*, pp. 1 – 4, Bordeau, September 2011.

[CSK12] N. Changuel, B. Sayadi, and M. Kieffer. Statistical multiplexing with distributed video coders control. *Submited to IEEE ICC*, 2012.

[CvdSsk10] N. Changuel, C. Mastronarde, M. van der Shaar, B. sayadi, and M. kieffer. End-to-end stochastic scheduling of scalable video over time-varying channels. *ACM Multimedia*, pp. 731 – 734, Florence, October 2010.

[CYOK99] H. Chi-Yuan, A. Ortega, and M. Khansari. Rate control for robust video transmission over burst-error wireless channels. *IEEE Journal on Selected Areas in Communication*, 17(5):756 – 773, May 1999.

[dCR03] P. de Cuetos and K. W. Ross. Optimal streaming of layered video: joint scheduling and error concealment. *ACM Multimedia*, pp. 2 – 8, Berkeley, November 2003.

[DOC+06] F. DeVito, T. Ozcelebi, M. Civanlar, A. Tekalp, and J.C. DeMartin. Per-GOP bitrate adaptation for H.264 compressed video sequences. *Visual Content Processing and Representation*, pp. 198 – 206, Berlin, September 2006. Springer-Verlag Berlin Heidelberg.

[Ell63] E. O. Elliott. Estimates of error rates for codes on burst-noise channels. *Bell Syst. Tech. J.*, 42:1977 – 1997, 1963.

[EM94] A.I. Elwalid and D. Mitra. Statistical multiplexing with loss priorities in rate-based congestion control of high-speed networks. *IEEE Transactions on Communications*, 42(11):2989 – 3002, November 1994.

[ESBa05] H. El-Sallabi, D. S. Baum, and al. Interim report on link level and system level channel models. Technical report, WINNER project internal deliverable D5.3, v.1.0, 2005.

[ETS05] ETSI. Multimedia Broadcast/Multicast service. (MBMS); UTRAN/GERAN requirements. Technical report, 3GPP TR 25.992-140, 2005.

[FL02] H. Fattah and C. Leung. An overview of scheduling algorithms in wireless multimedia networks. *IEEE Wireless Communications*, 9(5):76 – 83, October 2002.

[FL04] W.A.C. Fernando and K.K. Loo. Abrupt and gradual scene transition detection in MPEG-4 compressed video sequences using texture and macroblock information. *International Conference on Image Processing*, vol. 3, pp. 1589 – 1592, Piscataway, October 2004.

[FvdS09a] F. Fu and M. van der Schaar. A new systematic framework for autonomous cross-layer optimization. *IEEE Transactions on Vehicular Technology*, 58(4):1887 – 1903, May 2009.

[FVdS09b] F. Fu and M. Van der Schaar. Structural solutions for cross-layer optimization of wireless multimedia transmission. *Computing Research Repository - CORR*, abs/0905.4, 2009.

[FVdS10] F. Fu and M. Van der Schaar. Structural solutions to dynamic scheduling for multimedia transmission in unknown wireless environments. *Multimedia Systems and Applications Papers in Computer and Information Science*, abs/1008.4406, August 2010.

[GHM11] A. H. C. Gosline, V. Hayward, and H. Michalska. Ineluctability of oscillations in systems with digital implementation of derivative feedback. *Automatica*, November 2011.

[Gil60] E. N. Gilbert. Capacity of a burst-noise channel. *Bell Syst. Tech. Journal*, 39(9):1253 – 1265, 1960.

[GN98] R.M. Gray and D.L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44(6):2325 – 2383, 1998.

[HAK04] M. Hassan, L. Atzori, and M. Krunz. Video transport over wireless channels: a cycle-based approach for rate control. *ACM Multimedia*, pp. 916 – 923, New york, October 2004.

[Har11] Harmonic. IP-based distributed statistical multiplexing solution. Technical report, 2011.

[HCC08] Z. He, W. Cheng, and X. Chen. Energy minimization of portable video communication devices based on power-rate-distortion optimization. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(5):596 – 608, May 2008.

[HM02] Z. He and S. K. Mitra. A linear source model and a unified rate control algorithm for DCT video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(11):970 – 982, November 2002.

[HM09] Y. Huang and S. Mao. Analysis and design of a proportional-integral rate controller for streaming videos. *IEEE Conference on Global Telecommunications*, pp. 5530 – 5535, Honolulu, November 2009.

[HMM09] Y. Huang, S. Mao, and S.F. Midkiff. A control-theoretic approach to rate control for streaming videos. *IEEE Transactions on Multimedia*, 11(6):1072 –1081, October 2009.

[HW08] Z. He and D. O. Wu. Linear rate control and optimum statistical multiplexing for H.264 video broadcast. *IEEE Transactions on Multimedia*, 10(7):1237 – 1249, November 2008.

[II03] ITU-T and ISO/IEC JTC 1. Advanced video coding for generic audiovisual services. Technical report, ITU-T Rec. H.264, and ISO/IEC 14496-10 AVC, 2003.

[ITU08] ITU-T. Objective perceptual multimedia video quality measurement in the presence of a full reference. Technical Report Rec. J.247, ITU-T Rec. J.247 (08/08), 2008.

[JBT+08] M. Jacobs, J. Barbarien, S. Tondeur, R. Van de Walle, T. Paridaens, and P. Schelkens. Statistical multiplexing using SVC. *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting*, pp. 1 – 6, March 2008.

[JKM+92] M.A. Jolfaei, D. Kreuer, O. Maly, U. Quernheim, and W. Kremer. Time variant models for satellite channel. *IEEE Vehicular Technology Conference*, vol. 1, pp. 143 – 146, May 1992.

[KAM05] N. Kamaci, Y. Altunbasak, and R.M. Mersereau. Frame bit allocation for the H.264/AVC video coder via cauchy-density-based rate and distortion models. *IEEE Transaction on Circuits and Systems for Video Technology*, 15(8):994 – 1006, August 2005.

[KE03] K.V. Katsikopoulos and S. E. Engelbercht. Markov decision process with delays and asynchronous cost collection. *IEEE transactions on automatic control*, 48(4):568 – 574, April 2003.

[KHZ+03] W. Kumwilaisak, Y.T. Hou, Qian Zhang, Wenwu Zhu, C.-C.J. Kuo, and Ya-Qin Zhang. A cross-layer quality-of-service mapping architecture for video delivery in wireless networks. *IEEE Journal on Selected Areas in Communications*, 21(10):1685 – 1698, December 2003.

[KK96] K.-B. Kim and H.-J. Kim. Back-pressure buffering scheme to improve the cell loss property on the output buffered atm switch. *IEEE Conference on Local Computer Networks*, pp. 242 – 248, October 1996.

[KLLZ06] X. Kang, J. Lan, L. Liu, and X. Zhuang. SNR-based bit allocation in video quality smoothing. *Advances in Multimedia Information Processing*, vol. 4261 of *Lecture Notes in Computer Science*, pp. 989 – 998. 2006.

[Koe02] R. Koenen. MPEG-4 overview. Technical report, ISO/IEC JTC1/SC29/WG11, 2002.

[KW05] J-W. Kim, S-M. Byun and C-S. Won. A scene change detection in H.264/AVC compression domain. *Advances in Multimedia Information Processing*, vol. 3768, pp. 1072 – 1082, Coree, November 2005.

[LCGK10] J. Liu, Y. Cho, Z. Guo, and J. Kuo. Bit allocation for spatial scalability coding of H.264/SVC with dependent rate-distortion analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(7):967 – 981, July 2010.

[LEP+03] C.E. Luna, Y. Eisenberg, T.N. Pappas, R. Berry, and A.K. Katsaggelos. Joint source coding and data rate adaptation for energy efficient wireless video streaming. *IEEE Journal on Selected Areas in Communications*, 21(10):1710 – 1720, December 2003.

[LGP+03] Z. Li, W. Gao, F. Pan, S. Ma, K. Pang Lim, G. Feng, X. Lin, S. Rahardja, H. Lu, and Y. Lu. Adaptive basic unit layer rate control for JVT. Technical report, JVT-H014, 2003.

[LHK09] X. Li, A. Hutter, and A. Kaup. One-pass frame level budget allocation in video coding using inter-frame dependency. *IEEE International Workshop on Multimedia Signal Processing*, pp. 1 – 6, Rio de Janeiro, October 2009.

[Lin97] L.J. Lin. *Video Bit-Rate Control with Spline Approximated Rate-Distortion Characteristics*. PhD thesis, Univ. Southern California, May 1997.

[LJSB04] G. Liebl, H. Jenkac, T. Stockhammer, and C. Buchner. Radio link buffer management and scheduling for wireless video streaming. *Packet Video Workshop*, pp. 255 – 277, Irvine, December 2004.

[LLS08] Y. Liu, Z.G. Li, and Y.G Soh. Rate control of H.264/AVC scalable extension. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(1):116 – 121, January 2008.

[LO98] J. Lin and A. Ortega. Bit-rate control using piecewise approximated rate-distortion characteristics. *IEEE Transaction on Circuits System for Video Technology*, 8(4):446 – 459, August 1998.

[LOHK08] X. Li, N. Oertel, A. Hutter, and A. Kaup. Rate distortion optimized frame level rate control for H.264/AVC. *European Signal Processing Conference*, pp. 25 – 29, Lausane, August 2008.

[LXLZ01] L. Luo, J. Xu, S. Li, and Z. Zhuang. Rate control with smoothed temporal distortion for a 3D embedded wavelet video coder. *Proc. IEEE Int. Conf. Info., Commu. & Sig. Process*, 2001.

[LZ09] L. Liu and X. Zhuang. A novel square root rate control algorithm for H.264/AVC encoding. *IEEE International Conference on Multimedia and Expo*, pp. 814 – 817, Cancun, June 2009.

[LZK+09] T-L. Lin, Y. Zhi, S. Kanumuri, P.C. Cosman, and A.R. Reibman. Perceptual quality based packet dropping for generalized video gop structures. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 781 – 784, Taiwan, April 2009.

[MAvdS+03] A. Munteanu, Y. Andreopoulos, M. van der Schaar, P. Schelkens, and J. Cornelis. Control of the distortion variation in video coding systems based on motion compensated temporal filtering. *International Conference on Image Processing*, vol. 2, pp. 61 – 64, Barcelona, September 2003.

[MGL05] S. Ma, W. Gao, and Y. Lu. Rate-distortion analysis for H.264/AVC video coding and its application to rate control. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(12):1533 – 1544, December 2005.

[MGO04] C. Martin, A. Geurtz, and B. Ottersten. File based mobile satellite broadcast systems: Error rate computation and QoS based design. *Proceedings IEEE Vehicular Technology Conference*, vol. 6, pp. 4017 – 4021. September 2004.

[MK09] E. Maani and A.K. Katsaggelos. Optimized bit extraction using distortion modeling in the scalable extension of h.264/AVC. *IEEE Transactions on Image Processing*, 18(9):2022 – 2029, September 2009.

[MK10] E. Maani and A.K. Katsaggelos. Unequal error protection for robust streaming of scalable video over packet lossy networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(3):407 – 416, March 2010.

[MMS03] M. Mandjes, D. Mitra, and W. Scheinhardt. Models of network access using feedback fluid queues. *Queueing Systems: Theory and Applications*, 44(4):365 – 398, August 2003.

[MO00] Z. Miao and A. Ortega. Optimal scheduling for streaming of scalable media. *Conference Record of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1357 – 1362, Pacific Grove, October 2000.

[Mot09] Motorola. Statistical multiplexing over IP StatmuxIP solution. Technical report, Motorola, 2009. XP002565902.

[Mul93] F. Muller. Distribution shape of two-dimensional DCT coefficients of natural images. *Electronics Letters*, 29(22):1935 – 1936, October 1993.

[MvdS09] N.H. Mastronarde and M. van der Schaar. Designing autonomous layered video coders. *Signal processing. Image communication ISSN 0923-5965*, 24(6):417 – 436, July 2009.

[MvdS10] N. Mastronarde and M. van der Schaar. Online reinforcement learning for dynamic multimedia systems. *IEEE Transactions on Image Processing*, 19(2):290 – 305, February 2010.

[MvdS11] N. Mastronarde and M. van der Schaar. Fast reinforcement learning for energy-efficient wireless communications. *IEEE Transactions on Signal Processing*, abs/1009.5(99), December 2011.

[MVHB04a] X. Minghui, A. Vetro, S. Huifang, and L. Bede. Rate-distortion optimized bit allocation for error resilient video transcoding. *International Symposium on Circuits and Systems*, vol. 3, pp. 945 – 948, Vancouver, May 2004.

[MVHB04b] X. Minghui, A. Vetro, S. Huifang, and L. Bede. Rate distortion optimized bit allocation for error resilient video transcoding. *IEEE International Symposium on Circuits and Systems*, vol. 5, pp. 945 – 948, Vancouver, May 2004.

[Nic09] R. W. D Nickalls. The quartic equation: invariants and eulers solution revealed. *The Mathematical Gazette*, 93(526):66 75, March 2009.

[NKWR07] O. Nemethova, W. Karner, C. Weidmann, and M. Rupp. Distortion-minimizing network-aware scheduling for UMTS video streaming. *invited paper at EUSIPCO*, Poznan, September 2007.

[ON07] Z. Orlov and M. C. Necker. Enhancement of video streaming QoS with active buffer management in wireless environments. *European Wireless Conference*, Paris, 2007.

[oRIB08] Association of Radio Industries and Businesses. Multimedia Broadcast/Multicast Service (MBMS) user services stage 1 (release 8). Technical report, 3GPP TS, 2008.

[PA95] M. Perkins and D. Arnstein. Statistical multiplexing of multiple MPEG-2 video programs in a single channel. *Society of Motion Picture and Television Engineers Journal*, 104(9):596 – 599, October 1995.

[PBM+10] L. Peraldo, E. Baccaglini, E. Magli, G. Olmo, R. Ansari, and Y. Yao. Slice-level rate-distortion optimized multiple description coding for H.264/AVC. *IEEE International Conference on Acoustics Speech and Signal Processing*, pp. 2330 – 2333, Dallas, March 2010.

[Pic07] A. Picasso, B.; Bicchi. On the stabilization of linear systems under assigned I/O quantization. *IEEE Transactions on Automatic Control*, 52(10):1994 – 2000, October 2007.

[RBG08] M. Rezaei, I. Bouazizi, and M. Gabbouj. Joint video coding and statistical multiplexing for broadcasting over DVB-H channels. *IEEE Transactions on Multimedia*, 10(8):1455 – 1464, December 2008.

[RBG09] M. Rezaei, I. Bouazizi, and M. Gabbouj. Implementing statistical multiplexing in DVB-H. *International Journal of Digital Multimedia Broadcasting*, pp. 15, April 2009. Article ID 261231.

[Say05] K. Sayood. *Introduction to data compression*. Elsevier, 2005.

[SB98] R.S Sutton and A.G Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[SC03] H. Schulzrinne and S. Casner. *RTP profile for audo and video conferences with minimal control*. The Internet Engineering Task Force, request for comments 3551 edition, July 2003.

[SCFJ03] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications. RFC 3550, Network Working Group, 2003.

[SH05] K.J. Ström and T. Hägglund. *Advanced PID Control*. USA: ISA, Research Triangle Park, 2005.

[SLS08] Huey-Min Sun, Yung-Chuan Lin, and LihChyun Shu. The impact of varying frame rates and bit rates on perceived quality of low/high motion sequences with smooth/complex texture. *Multimedia Systems*, 14(1):1 – 13, May 2008.

[SMW07] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the scalable video coding extension of the H.264/AVC standard. *IEEE Trans. on Circuits and Systems for Video Technology In Circuits and Systems for Video Technology,*, 17(9):1103 – 1120, September 2007.

[Son78] E. J. Sondik. The optimal control of partially observable markov processes over the infinite horizon: Discounted cost. *Operations Research*, 26(2):282 – 304, March 1978.

[SRH10] M. Sablatschan, M. Ransburg, and H. Hellwagner. Towards an improved SVC-to-AVC rewriter. *International Conferences on Advances in Multimedia*, pp. 18 – 21, Los Alamitos, June 2010. IEEE Computer Society.

[SSWL06] H. Shen, X. Sun, F. Wu, and S. Li. Rate-distortion optimization for fast hierarchical B-picture transcoding. *IEEE International Symposium on Circuits and Systems*, vol. 5, pp. 5279 – 5282, Island of Kos, May 2006.

[STB09] S. Sesia, I. Toufik, and M. Baker. *LTE- The UMTS Long Term Evolution: From Theory to Practice*, chapter User Plane Protocols-Medium Access Control (MAC). John Wiley & Sons, 2009.

[SVHR10] S. K. Srinivasan, J. Vahabzadeh-Hagh, and M. Reisslein. The effects of priority levels and buffering on the statistical multiplexing of single-layer H.264/AVC and SVC encoded video streams. *IEEE Transactions on Broadcasting*, 56(3):281 – 287, September 2010.

[SW98] G. J. Sullivan and T. Wiegand. Rate-distortion optimization for video compression. *IEEE Signal Processing Magazine*, 15(6):74 – 90, November 1998.

[Ta10] Srisakul Thakolsri and Wolfgang Kellerer and. QoE-based rate adaptation scheme selection for resource-constrained wireless video transmission. *Proceedings of the International Conference on Multimedia*, pp. 783 – 786, Florence, October 2010. ACM.

[TC01] D.S Turaga and T. Chen. Hierarchical modeling of variable bit rate video sources. *Packet video Workshop*, Kyongju, April 2001.

[TDP00] B. Tao, B.W. Dickinson, and H.A. Peterson. Adaptive model-driven bit allocation for MPEG video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(1):147 – 157, February 2000.

[TVT08] M. Tagliasacchi, G. Valenzise, and S. Tubaro. Minimum variance optimal rate allocation for multiplexed H.264/AVC bitstreams. *IEEE Transactions on Image Processing*, 17(7):1129 – 1143, July 2008.

[V5.05] 3GPP TS 25.322 V5.12.0. Radio link control (RLC) protocol specification (release 5). Technical report, 3GPP, 2005.

[V7.09] 3GPP TS 25.322 V7.10.0. Radio link control (RLC) protocol specification (release 7). Technical report, 3GPP, 2009.

[vdZ08] G. M. van der Zalm. *Examination board tuning of PID type controllers: Literature overview*. Eindhoven : Technische Universiteit Eindhoven, 2008.

[VH08] V. Vukadinovic and J. Huschke. Statistical multiplexing gains of H.264/AVC video in E-MBMS. *International Symposium on Wireless Pervasive Computing*, pp. 468 – 474, Santorini, May 2008.

[VTT07] G. Valenzise, M. Tagliasacchi, and S. Tubaro. A smoothed, minimum distortion-variance rate control algorithm for multiplexed transcoded video sequences. *Proceedings of the International Workshop on Mobile*, pp. 55 – 60, Augsburg, September 2007. ACM.

[VWS07] J. Vieron, M. Wien, and H Schwarz. *JSVM9 software,*. Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG,, in 22nd JVT Meeting, No. JVT-V203, Marrakech, Morocco, Jan 2007.

[WB06] Z. Wu and J.M. Boyce. An error concealment scheme for entire frame losses based on H.264/AVC. *IEEE International Symposium on Circuits and Systems*, pp. 1 – 4, Island of Kos, May 2006.

[WCdWGA07] E. Witrant, C. Canudas-de Wit, D. Georges, and M. Alamir. Remote stabilization via communication networks with a distributed control law. *IEEE Transactions on Automatic Control*, 52(8):1480 – 1485, August 2007.

[WCSQ97] L. Wing-Cheong and L. San-Qi. Statistical multiplexing and buffer sharing in multimedia high-speed networks: a frequency-domain perspective. *IEEE/ACM Trans. on Networking*, 5(3):382 – 396, June 1997.

[Win05] G. Winkler. *Image Analysis, Random Fields and Dynamic Monte Carlo Methods: A Mathematical Introduction, Second edition*, vol. 27 of Applications of Mathematics. Springer, second edition, 2005.

[WK07] H. Wang and S. Kwong. A rate-distortion optimization algorithm for rate control in H.264. *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, pp. 1149 – 1152, Honolulu, May 2007.

[WM95] H. S. Wang and N. Moayeri. Finite-state markov channel - a useful model for radio communication channels. *IEEE transactions on vehicular technology*, 44(1):163 – 171, February 1995.

[WNLL09] T. J. Walsh, A. Nouri, L. Li, and Littman M. L. Learning and planning in environments with delayed feedback. *Autonomous Agents and Multi-Agent Systems*, 18(1):83 – 105, August 2009.

[WSBL03] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):560 – 576, July 2003.

[WSJ+03] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G.J. Sullivan. Rate-constrained coder control and comparison of video coding standards. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):688 – 703, July 2003.

[WSMG08] Y. Wang, J. Sun, S. Ma, and W. Gao. Theoretic analysis of inter frame dependency in video coding. *PCM '08*, pp. 935 – 939, Berlin, Heidelberg, 2008. Springer-Verlag.

[WV96] L. Wang and A. Vincent. Joint rate control for multi-program video coding. *IEEE Transactions on Consumer Electronics*, 42(3):300 – 305, August 1996.

[WZ09] X. Wang and Y. Zhang. Optimal video stream multiplexing in MBSFN. *IEEE International Conference on Communications Technology and Applications*, pp. 360 – 365, Beijing, October 2009.

[XMZG05] L. Xu, S. Ma, D. Zhao, and W. Gao. Rate control for scalable video model. *Visual Communications and Image Processing*, vol. 5960, pp. 525 – 534, Beijing, March 2005.

[YAB09] S. Yerima and K. Al-Begain. Dynamic buffer management for multimedia qos in beyond 3G wireless networks. *International Journal of Computer Science*, 36(4):378 – 387, November 2009.

[YDXD05] J. Yang, Q. Dai, W. Xu, and R. Ding. A rate control algorithm for MPEG-2 to H.264 real-time transcoding. *Visual Communications and Image Processing*, 5960:1995 – 2003, 2005.

[YW07] H. Yang and Q. Wang. A novel macroblock layer rate control for H.264/AVC. *IEEE International Conference on Multimedia and Expo*, pp. 432 – 435, Beijing, July 2007.

[ZAPS+07] X. Zhu, P. Agrawal, J. Pal Singh, T. Alpcan, and B. Girod. Rate allocation for multi-user video streaming over heterogenous access networks. *ACM Multimedia*, pp. 37 – 46, Augsburg, September 2007.

[ZK99] Q. Zhang and S.A. Kassam. Finite-state markov model for rayleigh fading channels. *IEEE Transactions on Communications*, 47(11):1688 – 1692, November 1999.

[ZNAT01] Z.L. Zhang, S. Nelakuditi, R. Aggarwal, and R.P. Tsang. Efficient selective frame discard algorithms for stored video delivery across resource constrained networks. *Real-Time Imaging*, 7(3):255 – 273, 2001.

[ZRJ91] J. Zdepski, D. Raychaudhuri, and K. Joseph. Statistically based buffer control policies for constant rate transmission of compressed digital video. *IEEE Transactions on Communications*, 39(6):947 – 957, June 1991.

Appendix A

Probability density function of E_j^0

The prediction residual

$$E_j^{X_j} = Y_j - \widehat{Y}_j = a_{X_j}\left(Y_{j-1} - q_{j-1}\left(Y_{j-1}\right)\right) + b_{X_j}U_j \tag{A.1}$$

is quantized with a step-size Δ_j. When $X_j = 0$, the quantized prediction residual corresponding to inter-coding is

$$E_j^0 = a_0\left(Y_{j-1} - q_{j-1}\left(Y_{j-1}\right)\right) + b_0 U_j. \tag{A.2}$$

The probability density function of E_j^0 is calculated as the convolution of the pdfs of U_j and of $a_0\left(Y_{j-1} - q_{j-1}\left(Y_{j-1}\right)\right)$. First, one aims at determining the pdf of $Y_{j-1} - q_{j-1}\left(Y_{j-1}\right)$. Assuming that the quantizations cells are of the form $[(k - 1/2)\Delta_j, (k + 1/2)\Delta_j[$ and that the reconstruction levels are of the form $k\Delta_j$, $k \in Z$ one gets that $I_j = Y_{j-1} - q_{j-1}\left(Y_{j-1}\right)$ remains in $[\Delta_j/2, \Delta_j/2[$ as

$$F_I(x) = \sum_{k=-\infty}^{+\infty} Pr[(k - 1/2)\Delta_j \leq I_j \leq k\Delta_j + x], \tag{A.3}$$

$$F_I(x) = \sum_{k=-\infty}^{+\infty} \left(F_Y(k\Delta_j + x) - F_Y((k - 1/2)\Delta_j)\right), \tag{A.4}$$

where F_Y is the cumulative distribution function of Y_j

$$F_I(x) = \sum_{k=-\infty}^{+\infty} \left(\int_{-\infty}^{k\Delta_j + x} f_Y(t)dt - \int_{-\infty}^{(k-1/2)\Delta_j} f_Y(t)dt \right). \tag{A.5}$$

Using the fact that the pdf f_Y of Y_j is

$$f_Y\left(x\right) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{x^2}{2\sigma_y^2}\right), \tag{A.6}$$

we get the pdf of I_j

$$f_I\left(x\right) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \sum_{k=-\infty}^{+\infty} \exp\left(-\frac{(x + k\delta_j)^2}{2\sigma_y^2}\right) \tag{A.7}$$

for all $x \in [-\Delta_j/2, \Delta_j/2[$. The pdf of E_j^0 is

$$f_{E_j^0}(x) = \int_{-\infty}^{+\infty} \frac{1}{a} f_I(\frac{t}{a}) f_u(x-t)dt \tag{A.8}$$

$$= \int_{-a\Delta_j/2}^{+a\Delta_j/2} \frac{1}{a} f_I(\frac{t}{a}) f_u(x-t)dt. \tag{A.9}$$

Since $f_I(x)$ is only defined over $[-\Delta_j/2, \Delta_j/2[$. Since

$$f_u(x) = \frac{1}{\sqrt{2\pi\sigma_u^2}} \exp\left(-\frac{x^2}{2\sigma_u^2}\right), \tag{A.10}$$

the probability distribution of E_j^0 becomes

$$f_{E_j^0}(x) = \frac{1}{\sqrt{8\pi\sigma_y^2}} \sum_{k=-\infty}^{+\infty} \exp\left(-\frac{(x + k\Delta_{j-1}a_0)^2}{2\sigma_y^2}\right) G(x, \Delta_{j-1}, k), \tag{A.11}$$

where

$$G(x, \Delta_{j-1}, k) = \operatorname{erf}\left(\frac{2a_0x + \Delta_{j-1}(1 - 2k(1-a_0^2))}{2\sqrt{2}\sigma_y\sqrt{1-a_0^2}}\right)$$
$$-\operatorname{erf}\left(\frac{2a_0x - \Delta_{j-1}(1 + 2k(1-a_0^2))}{2\sqrt{2}\sigma_y\sqrt{1-a_0^2}}\right). \tag{A.12}$$

Appendix B

Stability of the proposed distributed SM systems

1 Reminder about the stability of a linear system

Consider the closed loop control system represented in Figure B.1. In this example,

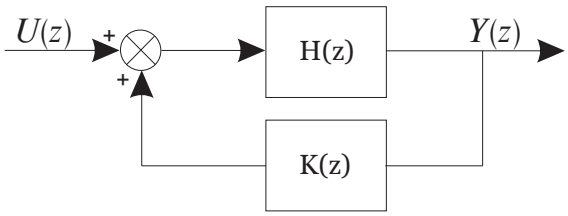


Figure B.1: Block diagram representation of the state space equations.

the stability of the closed-loop transfer function

$$Y(z) = \frac{H(z)}{1 + H(z)K(z)} U(z)$$

depends on the location of the roots of the equation $1 + H(z)K(z) = 0$. Consider now the state-space representation

$$\begin{aligned} X(j+1) &= AX(j) + BU(j) \\ Y(j) &= CX(j) \end{aligned} \tag{B.1}$$

of a Multi-Input-Multi-Output (MIMO) linear system with $U(z)$ as inputs, $Y(z)$ as outputs, and $X(z)$ the vector of state variables. In (B.1), A is the state matrix, B is the input matrix, C is the output matrix, and j is the temporal index. The stability of this system depends on the location of the poles of the transfer function

$$H(z) = C(zI - A)^{-1} B.$$

These poles are the zeros of

$$f(z) = \det(zI - A).$$

2 Control of the buffer level in bits

Here, we study the stability of the proposed distributed statistical multiplexing system using a PI controller to adjust the encoding rate at each video encoder and a PI for the bandwidth allocation. To derive the stability analysis of the control system, one should first represent it as a discrete-time state-space representation as follows

$$\begin{aligned} X^{\mathrm{b}}(j+1) &= A^{\mathrm{b}}X^{\mathrm{b}}(j) + B^{\mathrm{b}}U^{\mathrm{b}}(j) \\ Y^{\mathrm{b}}(j) &= C^{\mathrm{b}}X^{\mathrm{b}}(j), \end{aligned}$$

with

$$\begin{aligned} X^{\mathrm{b}}(j) = \Big(&\tilde{\beta}_1(j), \ldots, \tilde{\beta}_N(j), \Delta_1(j), \ldots, \Delta_N(j), \phi_1(j), \ldots, \phi_N(j), \\ &\tilde{P}_1(j), \ldots, \tilde{P}_N(j), \tilde{B}_1(j), \ldots, \tilde{B}_N(j)\Big)^T, \end{aligned} \tag{B.2}$$

and

$$U^{\mathrm{b}}(j) = \left(\delta\beta_1(j), \ldots, \delta\beta_N(j)\right)^T. \tag{B.3}$$

In the considered system, we assume that only the state of the buffer and the PSNR of the encoded VUs may be measured, thus, the state matrix A^{b}, the input matrix B^{b}, and the output matrix C^{b} are

$$A^{\mathrm{b}} = \begin{pmatrix} I & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & I \\ 0 & 0 & I & -L & 0 \\ I & -\gamma K_I^{bR} I & 0 & 0 & -\gamma\left(K_P^{bR} + K_I^{bR}\right) I \\ 0 & -K_I^R I & -K_I^\theta T I & \left(K_P^\theta + K_I^\theta\right) TL & \left(1 - \left(K_P^{bR} + K_I^{bR}\right)\right) I \end{pmatrix} \tag{B.4}$$

$$B^{\mathrm{b}} = \begin{pmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{B.5}$$

and

$$C^{\mathrm{b}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{pmatrix}, \tag{B.6}$$

with

$$L = \frac{1}{N}\begin{pmatrix} N-1 & -1 & -1 \\ -1 & \ddots & -1 \\ -1 & -1 & N-1 \end{pmatrix},$$

and

$$\text{zI-A}^{\text{b}} = \begin{pmatrix} (z-1)\,I & 0 & 0 & 0 & 0 \\ 0 & (z-1)\,I & 0 & 0 & -I \\ 0 & 0 & (z-1)\,I & L & 0 \\ -I & \gamma K_I^{bR} I & 0 & zI & \gamma\left(K_P^{bR}+K_I^{bR}\right)I \\ 0 & K_I^R I & K_I^\theta T I & -\left(K_P^\theta+K_I^\theta\right)TL & \left(z-1+\left(K_P^{bR}+K_I^{bR}\right)\right)I \end{pmatrix}. \tag{B.7}$$

Using the matrix property

$$\det\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = \det A_{11}\det\left(A_{22}-A_{21}A_{11}^{-1}A_{12}\right), \tag{B.8}$$

one can develop $f(z)=\det\left(zI-A^{\text{b}}\right)$ as follows

$$\begin{aligned} f(z) &= (z-1)^{3N}\det\left(\begin{pmatrix} zI & \gamma\left(K_P^{bR}+K_I^{bR}\right)I \\ -\left(K_P^\theta+K_I^\theta\right)TL & \left(z-1+\left(K_P^{bR}+K_I^{bR}\right)\right)I \end{pmatrix} - (z-1)^{-1}\begin{pmatrix} 0 & -\gamma K_I^{bR} I \\ K_I^\theta TL & -K_I^{bR} I \end{pmatrix}\right) \\ &= (z-1)^{3N}\det\begin{pmatrix} zI & \gamma\left(K_P^{bR}+K_I^{bR}\right)I+(z-1)^{-1}\gamma K_I^{bR}I \\ -\left(K_P^\theta+K_I^\theta\right)TL-(z-1)^{-1}K_I^\theta TL & \left(z-1+\left(K_P^{bR}+K_I^{bR}\right)\right)I+(z-1)^{-1}K_I^{bR}I \end{pmatrix} \\ &= (z-1)^{3N}\det(F), \end{aligned} \tag{B.9}$$

with

$$\text{F}=\begin{pmatrix} zI & \gamma\left(K_P^{bR}+K_I^{bR}\right)I+(z-1)^{-1}\gamma K_I^{bR}I \\ -\left(K_P^\theta+K_I^\theta\right)TL-(z-1)^{-1}K_I^\theta TL & \left(z-1+\left(K_P^{bR}+K_I^{bR}\right)\right)I+(z-1)^{-1}K_I^{bR}I \end{pmatrix}. \tag{B.10}$$

$$\begin{aligned} \det(F) &= \det(zI)\det(z^{-1}(z-1)^{-2}(\left(z\,(z-1)^3+\left(K_P^{bR}+K_I^{bR}\right)z\,(z-1)^2+K_I^{bR}\,(z-1)\,z\right)I \\ &\quad -\left(-\left(K_P^\theta+K_I^\theta\right)T\,(z-1)\,L-K_I^\theta TL\right)\left(\gamma\left(K_P^{bR}+K_I^{bR}\right)(z-1)\,I+\gamma K_I^{bR}I\right)) \end{aligned} \tag{B.11}$$

$$\begin{aligned} \det(F) &= (z-1)^{-2N}\det\left(a(z)I+b(z)L\right) \\ &= (z-1)^{-2N}\det\left(\left(a_1z^4+a_2z^3+a_3z^2+a_4z+a_5\right)I+\left(a_6z^2+a_7z+a_8\right)L\right), \end{aligned} \tag{B.12}$$

with

$$\begin{aligned} a(z) &= a_1z^4+a_2z^3+a_3z^2+a_4z+a_5 \\ b(z) &= b_1z^2+b_2z+b_3 \end{aligned} \tag{B.13}$$

and

$$\begin{cases} a_1 = 1 \\ a_2 = -3+\left(K_P^{bR}+K_I^{bR}\right) \\ a_3 = 3-2\left(K_P^{bR}+K_I^{bR}\right)+K_I^{bR} \\ a_4 = -1+\left(K_P^{bR}\right. \\ a_5 = 0 \\ b_1 = \gamma\left(K_P^{bR}+K_I^{bR}\right)\left(K_P^\theta+K_I^\theta\right)T \\ b_2 = -2\gamma K_P^{bR}\left(K_P^\theta+K_I^\theta\right)T+\gamma K_P^{bR}\left(K_I^\theta+K_P^\theta\right)T+\gamma K_I^{bR}K_P^\theta T \\ b_3 = \gamma K_I^{bR}K_I^\theta T-\gamma K_I^{bR}\left(K_P^\theta+K_I^\theta\right)T+\gamma\left(K_P^{bR}+K_I^{bR}\right)K_P^\theta T \end{cases} \tag{B.14}$$

Thus,

$$\begin{aligned} f(z) &= (z-1)^N \det\left(a(z)I + b(z)L\right) \\ &= (z-1)^N \left(\tfrac{b(z)}{N}\right)^N \det M, \end{aligned} \tag{B.15}$$

with

$$M = \begin{pmatrix} N-1+\frac{Na(z)}{b(z)} & -1 & -1 \\ -1 & \ddots & -1 \\ -1 & -1 & N-1+\frac{Na(z)}{b(z)} \end{pmatrix}.$$

Since M is a circulant matrix, its determinant is

$$\det M = \prod_{i=1}^{N} \left(N-1+\frac{Na(z)}{b(z)} - \omega_i - \omega_i^2 - \cdots - \omega_i^{N-1}\right),$$

because

$$\omega_i + \omega_i^2 + \cdots + \omega_i^{N-1} = -1 + \frac{1-\omega_i^N}{1-\omega_i} = -1,$$

where $\omega_i = \exp\left(\frac{2\pi i\sqrt{-1}}{N}\right)$. We get

$$\det M = \left(N + \frac{Na(z)}{b(z)}\right)^N. \tag{B.16}$$

Finally

$$\begin{aligned} f(z) &= (z-1)^N \left(a(z)+b(z)\right)^N \\ &= (z-1)^N \left(c_1 z^4 + c_2 z^3 + c_3 z^2 + c_4 z + c_5\right)^N, \end{aligned} \tag{B.17}$$

with

$$\left\{\begin{array}{rcl} c_1 &=& 1 \\ c_2 &=& -3 + \left(K_P^{bR} + K_I^{bR}\right) \\ c_3 &=& 3 - 2\left(K_P^{bR} + K_I^{bR}\right) + K_I^{bR} + \gamma\left(K_P^{bR} + K_I^{bR}\right)\left(K_P^{\theta} + K_I^{\theta}\right)T \\ c_4 &=& -1 + K_P^{bR} - 2\gamma K_P^{bR}\left(K_P^{\theta} + K_I^{\theta}\right)T + \gamma K_P^{bR}\left(K_I^{\theta} + K_P^{\theta}\right)T + \gamma K_I^{bR} K_P^{\theta} T \\ c_5 &=& \gamma K_I^{bR} K_I^{\theta} T - \gamma K_I^{bR}\left(K_P^{\theta} + K_I^{\theta}\right)T + \gamma\left(K_P^{bR} + K_I^{bR}\right)K_P^{\theta} T \end{array}\right. \tag{B.18}$$

If a P controllers are used in the rate allocation and the encoder control, thus $K_I^{bR} = 0$ and $K_I^{\theta} = 0$ and

$$\left\{\begin{array}{rcl} c_1 &=& 1 \\ c_2 &=& -3 + K_P^{bR} \\ c_3 &=& 3 - 2K_P^{bR} + \gamma K_P^{bR} K_P^{\theta} T \\ c_4 &=& -1 + K_P^{bR} - 2\gamma K_P^{bR} K_P^{\theta} T \\ c_5 &=& \gamma K_P^{bR} K_P^{\theta} T \end{array}\right., \tag{B.19}$$

leading to

$$\begin{aligned} f(z) &= (z-1)^N \left(c_1 z^4 + c_2 z^3 + c_3 z^2 + c_4 z + c_5\right)^N \\ & (z-1)^N (z-1)^{2N} \left(z^2 - \left(1 - K_P^{bR}\right) z + \gamma K_P^{bR} K_P^{\theta} T\right)^N. \end{aligned} \tag{B.20}$$

3 Control of the buffer level in seconds

Here we consider stability analysis when buffer level in seconds is controlled using PI controllers in both bandwidth allocation and encoder control. To derive the stability analysis of the control system, one should first represent it as a discrete-time state-space representation as follows

$$\begin{array}{l} \Delta X^{\tau}(j+1) = A^{\tau}\Delta X^{\tau}(j) + B^{\tau}\Delta U^{\tau}(j) \\ \Delta Y^{\tau}(j) = C^{\tau}\Delta X^{\tau}(j), \end{array} \tag{B.21}$$

with

$$\begin{aligned} \Delta X^{\tau}(j) = \Big(&\Delta\tilde{\beta}_1(j), \ldots, \Delta\tilde{\beta}_N(j), \Delta\Pi_1(j), \ldots \Delta\Pi_N(j), \Delta\phi_1(j), \ldots, \Delta\phi_N(j), \\ &\Delta\tilde{R}_1^{\mathrm{e}}(j), \ldots, \Delta\tilde{R}_N^{\mathrm{e}}(j), \tilde{P}_1(j), \ldots, \Delta\tilde{P}_N(j), \Delta B_1(j), \ldots, \Delta B_N(j)\Big)^{T}, \end{aligned} \tag{B.22}$$

and

$$\Delta U^{\tau}(j) = \left(\delta\beta_1(j) \ldots \delta\beta_N(j)\right)^{T}. \tag{B.23}$$

In the considered system, we assume that the state of the buffer, the encoding rate, and the PSNR of the encoded VUs may be measured, thus one can deduce the state matrix A^{τ}, the input matrix B^{τ}, and the output matrix C^{τ}

$$\mathrm{A}^{\tau} = \begin{pmatrix} I & 0 & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & -L & 0 \\ 0 & 0 & I & -\tau_0 VI & 0 & VI \\ 0 & 0 & -\alpha\frac{K_I^{\tau R}}{T}I & \left(\alpha\left(\frac{K_P^{\tau R}}{T}+\frac{K_I^{\tau R}}{T}\right)\tau_0 V + 1 - \alpha\right)I & 0 & -\alpha\left(\frac{K_P^{\tau R}}{T}+\frac{K_I^{\tau R}}{T}\right)VI \\ I & 0 & -\gamma K_I^{\tau R} I & \gamma\tau_0 V\left(K_P^{\tau R}+K_I^{\tau R}\right)I & 0 & -\gamma V\left(K_P^{\tau R}+K_I^{\tau R}\right)I \\ 0 & -K_I^{\theta} TI & -K_I^{\tau R} I & \left(K_P^{\tau R}+K_I^{\tau R}\right)\tau_0 VI & \left(K_P^{\theta}+K_I^{\theta}\right)TL & \left(1-\left(K_P^{\tau R}+K_I^{\tau R}\right)V\right)I \end{pmatrix}, \tag{B.24}$$

$$B^{\tau} = \begin{pmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{B.25}$$

and

$$C^{\mathrm{b}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{pmatrix}, \tag{B.26}$$

with $V = \frac{1}{\bar{R}_i^{\mathrm{eq}}}$.

The stability of this system depends on the location of the poles of the transfer function $C^{\tau}\left(zI - A^{\tau}\right)^{-1}B^{\tau}$.

These poles are the zeros of

$$g(z) = \det\left(zI - A^{\tau}\right).$$

$$\text{zI-A}^{\tau} = \begin{pmatrix} (z-1)\,I & 0 & 0 & 0 & 0 & 0 \\ 0 & (z-1)\,I & 0 & 0 & L & 0 \\ 0 & 0 & (z-1)\,I & \tau_0 V I & 0 & -VI \\ 0 & 0 & \alpha\frac{K_I^{\tau R}}{T} I & \left(z-\alpha\left(\frac{K_P^{\tau R}}{T}+\frac{K_I^{\tau R}}{T}\right)\tau_0 V-1+\alpha\right) I & 0 & \alpha\left(\frac{K_P^{\tau R}}{T}+\frac{K_I^{\tau R}}{T}\right) VI \\ -I & 0 & \gamma K_I^{\tau R} I & -\gamma\tau_0 V\left(K_P^{\tau R}+K_I^{\tau R}\right) I & zI & \gamma V\left(K_P^{\tau R}+K_I^{\tau R}\right) I \\ 0 & K_I^{\theta} TI & K_I^{\tau R} I & -\left(K_P^{\tau R}+K_I^{\tau R}\right)\tau_0 VI & -\left(K_P^{\theta}+K_I^{\theta}\right) TL & \left(z-1+\left(K_P^{\tau R}+K_I^{\tau R}\right) V\right) \end{pmatrix}. \tag{B.27}$$

Using the matrix property

$$\det\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = \det A_{11}\det\left(A_{22}-A_{21}A_{11}^{-1}A_{12}\right), \tag{B.28}$$

one can develop $g(z)=\det\left(zI-A^{\tau}\right)$ as follows

$$g(z) = (z-1)^{3N}\det\left(D-(z-1)^{-1}M\right) = (z-1)^{2N}\det\left(F\right), \tag{B.29}$$

with

$$D = \begin{pmatrix} \left(z-\alpha\frac{K^{\tau R}}{T}\tau_0 V-1+\alpha\right) I & 0 & \alpha\frac{K^{\tau R}}{T} VI \\ -\gamma\tau_0 V K^{\tau R} I & zI & \gamma V K^{\tau R} I \\ -K^{\tau R}\tau_0 VI & -K_I^{\theta} TL & \left(z-1+K^{\tau R} V\right) \end{pmatrix}, \tag{B.30}$$

$$M = \begin{pmatrix} \tau_0\alpha\frac{K_I^{\tau R}}{T} V & 0 & -\alpha\frac{K_I^{\tau R}}{T} VI \\ \gamma K_I^{\tau R}\tau_0 VI & 0 & -V\gamma K_I^{\tau R} I \\ K_I^{\tau R}\tau_0 VI & K_I^{\theta} TL & -VK_I^{\tau R} \end{pmatrix}, \tag{B.31}$$

and

$$\text{F}=\begin{pmatrix} P_1(z)I & 0 & \left(\alpha\frac{K^{\tau R}}{T} V\,(z-1)+\alpha\frac{K_I^{\tau R}}{T} V\right) I \\ \left(-\gamma\tau_0 V K^{\tau R}\,(z-1)-\gamma K_I^{\tau R}\tau_0 V\right) I & z\,(z-1)\,I & \left(\gamma V K^{\tau R}\,(z-1)+\gamma K_I^{\tau R} V\right) I \\ \left(-K^{\tau R}\tau_0 V\,(z-1)-K\tau_0 V\right) I & -K_I^{\theta} T\,(z-1)\,L-K_I^{\theta} TL & \left(\left(z-1+K^{\tau R} V\right)(z-1)+K_I^{\tau R} V\right) I \end{pmatrix}, \tag{B.32}$$

with $K^{\tau R}=K_P^{\tau R}+K_I^{\tau R}$, $K^{\theta}=K_P^{\theta}+K_I^{\theta}$, and $P_1(z)=a_1z^2+a_2z+a_3$ with

$$\begin{cases} a_1 = & 1 \\ a_2 = & -\alpha\frac{K^{\tau R}}{T}\tau_0 V-2+\alpha \\ a_3 = & \alpha\frac{K^{\tau R}}{T}\tau_0 V+1-\alpha-\alpha\tau_0 V\frac{K_I^{\tau R}}{T} \end{cases} \tag{B.33}$$

Then,

$$\begin{aligned} \det\left(F\right) &= \left(P_1(z)\right)^N\det\left(D'-\left(az^2+bz+c\right)^{-1}M'\right) && \text{(B.34)} \\ &= \left(P_1(z)\right)^N\det\left(F'\right), && \text{(B.35)} \end{aligned}$$

with

$$D' = \begin{pmatrix} z\,(z-1)\,I & \left(\gamma V K^{\tau R}\,(z-1)+\gamma K_I^{\tau R} V\right) I \\ -K_I^{\theta} T\,(z-1)\,L-K_I^{\theta} TL & \left(\left(z-1+K^{\tau R} V\right)(z-1)+K_I^{\tau R} V\right) I \end{pmatrix}, \tag{B.36}$$

$$M' = \begin{pmatrix} 0 & \left(-\gamma\tau_0 V K^{\tau R}(z-1) - \gamma K_I^{\tau R}\tau_0 V\right)\left(\alpha\frac{K^{\tau R}}{T}V(z-1) + \alpha\frac{K_I^{\tau R}}{T}V\right) \\ 0 & \left(-K^{\tau R}\tau_0 V(z-1) - K_I^{\tau R}\tau_0 V\right)\left(\alpha\frac{K^{\tau R}}{T}V(z-1) + \alpha\frac{K_I^{\tau R}}{T}V\right) \end{pmatrix}, \tag{B.37}$$

$$F' = P_1(z)^{-1}\begin{pmatrix} P_2(z)I & P_3(z)I \\ P_4(z)L & P_5(z)I \end{pmatrix}, \tag{B.38}$$

and

$$\begin{cases} P_2(z) = & a_1 z^4 + (a_2 - a_1)z^3 + (a_3 - a_2)z^2 - a_3 z \\ P_3(z) = & \gamma V K^{\tau R} a_1 z^3 + \gamma V\left(a_2 K^{\tau R} - K_P^{\tau R} a_1 + \frac{\tau_0 V K^{\tau R}\alpha K^{\tau R}}{T}\right)z^2 \\ & +\gamma V\left(K^{\tau R}a_3 - K_P^{\tau R}a_2 - 2\frac{\alpha K_P^{\tau R}V^2\gamma\tau_0 K^{\tau R}}{T}\right)z + \frac{\gamma V\alpha V\tau_0 K_P^{\tau R}K_P^{\tau R}}{T} - K_P^{\tau R}a_3 \\ P_4(z) = & -K_I^{\theta}Ta_1 z^3 - \left(2K_I^{\theta}Ta_1 + a_2 K_I^{\theta}T\right)z^2 - \left(2K_I^{\theta}Ta_2 + K_I^{\theta}Ta_3\right)z - 2K_I^{\theta}Ta_3 \\ P_5(z) = & a_1 z^4 + \left(a_1\left(K^{\tau R}V - 2\right) + a_2\right)z^3 + \left(a_1\left(1 - K_P^{\tau R}V\right) + a_2\left(K^{\tau R}V - 2\right) + a^3 + \frac{\tau_0 V^2\alpha}{T}K^{\tau R}K^{\tau R}\right)z^2 \\ & + \left(\left(1 - K_P^{\tau R}V\right)a_2 + \left(K^{\tau R}V - 2\right)a_3 - 2\frac{\tau_0 V^2\alpha}{T}K^{\tau R}K_P^{\tau R}\right)z + 1 - K_P^{\tau R}Va_3 + K_P^{\tau R}K_P^{\tau R}\frac{\tau_0 V^2\alpha}{T} \end{cases} \tag{B.39}$$

Finally

$$\begin{aligned} \text{g(z)=(z-1)}^{2N}\det & \begin{pmatrix} P_2(z)I & P_3(z)I \\ P_4(z)L & P_5(z)I \end{pmatrix}. \\ &= (z-1)^{2N}\det\left(P_2(z)I\right)\det\left(P_5(z)I - \left(P_2(z)I\right)^{-1}P_3(z)IP_4(z)L\right) \\ &= (z-1)^{2N}\det\left(P_5(z)P_2(z)I - P_3(z)P_4(z)L\right) \\ &= (z-1)^{2N}\det\left(P_5(z)P_2(z)I - \tfrac{1}{N}P_3(z)P_4(z)\begin{pmatrix} N-1 & -1 & -1 \\ -1 & \ddots & -1 \\ -1 & -1 & N-1 \end{pmatrix}\right) \\ &= (z-1)^{2N}\det\left(-\tfrac{1}{N}P_3(z)P_4(z)\begin{pmatrix} N-1-\frac{NP_5(z)P_2(z)}{P_3(z)P_4(z)} & -1 & -1 \\ -1 & \ddots & -1 \\ -1 & -1 & N-1-\frac{NP_5(z)P_2(z)}{P_3(z)P_4(z)} \end{pmatrix}\right). \end{aligned} \tag{B.38}$$

Using properties of circulant matrix, one gets

$$g(z) = (z-1)^{2N}\left(P_5(z)P_2(z) - P_3(z)P_4(z)\right)^N. \tag{B.39}$$

Appendix C

State transition probabilities

We focus on the evaluation of state transition probability for all considered states in Chapter 8 in order to define the state transition probability P.

1 System states

1.1 Channel state

The channel state h_t represents the channel condition in time slot $[t, t+1)$ assumed constant. The changes of the wireless channel state h is modeled as an N_h-state Markov chain to simulate the bursty nature of an error-prone wireless channel with the state transition probability given by $p(h_{t+1}|h_t)$. The channel state transition probabilities are described by

$$p_{ij} = p(h_{t+1} = i \,|\, h_t = j), \text{ with } i, j \in \{1, \ldots, N_h\}\,. \tag{C.1}$$

These probabilities are estimated and learned on-line.

1.2 Frame state

The frame state s^I transition probabilities can be defined from the GoP structure assumed constant for all the sequence. N_G is the size of the GoP set by the encoder and assumed constant with time. A 3-state Markov chain can be used to model the transitions between the three frame types (I,P, and B). The frame state transition probabilities depend on the desired ratio of I, P, and B data units as well as the GoP structure (e.g. IBPBPBP, IPBBPBB, etc.) set by the video coder. Consider an example of GoP structure (IPPPPPP) with only I and P frames.

A 2-states Markov chain can be used to model the transitions between the two frame types (I and P). The probability of moving from a P frame to an other P frame is $p_{PP} = p1 = \frac{N_G-2}{N_G-1}$ and the probability of moving from a P frame to an I frame in the GoP is $p_{PI} = 1 - p1 = \frac{1}{N_G-1}$, with $p_{II} = 0$ and $p_{IP} = 1$.

1.3 Post-encoder buffer

At each time slot, a new frame encoded with L SNR layers is stored in the post-encoder buffer which can hold a maximum of B^e frames. We assume that post-encoder buffers are first-in first-out (FIFO) queues. The buffer state $s^e \in \mathcal{S}^e =$

$\{0, 1, \ldots, B^e\}$ corresponds to the number of frames stored in the buffers and it evolves as follows

$$\begin{aligned} s_0^e = &\quad s_{int}^e, \\ s_{t+1}^e = &\quad \max(0, \min(s_t^e + 1 - |\max(a_t(1:L))| \\ &\quad -|\max(a_t(L+1:2L))|, B^e)), \end{aligned}$$

where s_{int}^e is the initial buffer state, $|\max(a_t(1:L))|$ is equal to 1 if at least one SNR layer of the oldest frame in the buffer is transmitted or dropped and to 0 otherwise, and $|\max(a_t(L+1:2L))|$ is equal to 1 if at least one SNR layer of the oldest frame but one is transmitted or dropped and to 0 otherwise.

1.4 MAC buffer

The packets sent by the post-encoder buffer are segmented in PDU packets. M is the size of each PDU packet. These packets are stored in a finite-length MAC buffer, which can hold a maximum of B^m PDU packets. The MAC buffer state $s^m \in \mathcal{S}^m = \{0, 1, \ldots, B^m\}$ evolves as follows

$$\begin{aligned} s_0^m = &\quad s_{int}^m, \\ s_{t+1}^m = &\quad \max(0, \min(s_{t+1}^m + \left\lceil \textstyle\sum_{k=1}^{L} a_t(2L+1-k) b_t(s_t^I, k) \right. \\ &\quad \left. + \textstyle\sum_{k=1}^{L} a_t(L+1-k) b_{t+1}(s_t^I, k) - R_t^c(h_t) \right\rceil \tfrac{1}{M}, B^m)), \end{aligned} \tag{C.2}$$

where s_{int}^m is the initial buffer state, R_t^c represents the available channel rate at time t and $\lceil . \rceil$ is the round toward $+\infty$ to represent the PDU packets segmentation.

2 Model state: system state space reduction

The considered system states would lead to a very large number of values taken by the state in the context of filtering of scalable layers. Q-learning algorithms used for the scheduling decision making would not converge in a reasonable amount of time. Thus, a coarser representation of the level of the state has to be considered. Model states are defined as a quantification of the already defined system state. Here, we detail how to quantify and reduce the system state space to define the model states. We also express the state transition probabilities of the new model states.

2.1 Frame state

The considered frame state space can be easily reduced. In fact, it is obvious that the I frame has a important impact on the video quality than the P or the B frame so it should be considered with higher priority level (*e.g,* avoid dropping action, more SNR layers to transmit). Thus, in order to reduce the state space, one can model the frame type by the position of the I frame among the two concerned frames.

Thus, one can consider the state $s^I \in \mathcal{S}^I$ as the position of the I frame among the different possible positions in the post-encoder buffer. $\mathcal{S}^I = \{0, 1, 2\}$ represents the sets of possible values taken by s^I. Examples of some frame type state are in Figure C.1.

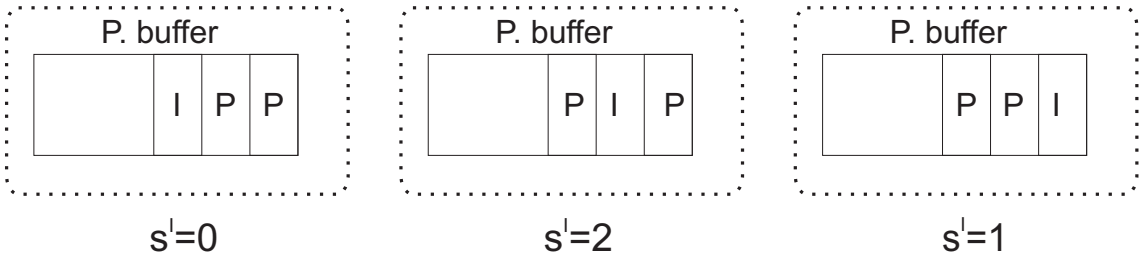


Figure C.1: Example of frame type position state.

The evolution of the state s_t^I depends on the previous state and the considered action at the post-encoder buffer $\mathbf{a}_t$. We consider three possibles positions for the I frame: the last frame in the post-encoder buffer $s_t^I = 1$, the last frame but one in the post-encoder buffer $s_t^I = 2$, and elsewhere $s_t^I = 0$. If $s_t^I \neq 0$

$$\begin{array}{ll} s_0^I = s_{int}^I, & \\ s_{t+1}^I = s_t^I + f(\mathbf{a}_t) & \text{if } s_t^I + f(\mathbf{a}_t) \leq 2, \\ s_{t+1}^I = 0 & \text{otherwise,} \end{array}$$

Where $f(.)$ is a function which serves to group actions into three class: $f(\mathbf{a}_t) = 0$ if no SNR layer is transmitted, $f(\mathbf{a}_t) = 1$ if at least one SNR layer of the oldest frame is transmitted or dropped, and $f(\mathbf{a}_t) = 2$ if at least one SNR layer of the oldest frame and one SNR layer of the following frame are transmitted from the buffer. In the case $s_t^I = 0$, the state transition depends on the frame type transition probability. We consider two frames types: I frame, and P frame. Consider an example of GoP structure (IPPPPPP) with only I and P frames.

A two-states Markov chain can be used to model the transitions between the two frame types (I and P). Lets $p_{PP} = p1 = \frac{N_G-2}{N_G-1}$ be the probability of moving from a P frame to an other P frame and $p_{PI} = 1 - p1 = \frac{1}{N_G-1}$ be the probability of moving from a P frame to an I frame in the GoP with $p_{II} = 0$ and $p_{IP} = 1$.

With these probabilities, one can define the I frame position state transition probability when $s_t^I = 0$

$$\begin{array}{l} p(s_{t+1}^I = 0 | s_t^I = 0, f(\mathbf{a}_t) = 0) = 1, \\ p(s_{t+1}^I = 0 | s_t^I = 0, f(\mathbf{a}_t) = 1) = p1, \\ p(s_{t+1}^I = 1 | s_t^I = 0, f(\mathbf{a}_t) = 1) = 1 - p1, \\ p(s_{t+1}^I = 0 | s_t^I = 0, f(\mathbf{a}_t) = 2) = p1 \times p1, \\ p(s_{t+1}^I = 1 | s_t^I = 0, f(\mathbf{a}_t) = 2) = p1 \times (1 - p1), \\ p(s_{t+1}^I = 2 | s_t^I = 0, f(\mathbf{a}_t) = 2) = 1 - p1. \end{array}$$

2.2 Buffer state

Concerning the post-encoder and the MAC buffers states, we propose to quantify the state space of the post-encoder buffer $\mathcal{S}^e = \{0, 1 \dots B^e\}$ into N_e intervals to model the post-encoder buffer state so that the MAC buffer states $\mathcal{S}^m = \{0, 1 \dots B^m\}$ are quantized into N_m intervals. We assume states to be uniformly distributed within the different intervals and so

$$p(s_{t+1} = j \in [j_1, j_2]) = \frac{1}{j_2 - j_1}, \tag{C.3}$$

and the quantized state transition probability is expressed as follow

$$p(s_{t+1} \in [i_1, i_2] | s_t \in [j_1, j_2], \mathbf{a}_t) = \frac{\sum_i \sum_j p(s_{t+1} = i | s_t = j, \mathbf{a}_t) p(s_t = j)}{\sum_j p(s_t = j)},$$

where $[i_1, i_2]$ and $[j_1, j_2]$ are two of the several intervals and i and j are elements in these intervals.

2.3 Channel state

Similarly as for the post-encoder and the MAC buffers, one can also reduce the system channel state space by quantizing the N_h Markov states into $N'_h < N_h$ Markov states. This will be at the price of a coarser granularity for the channel state model.